

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

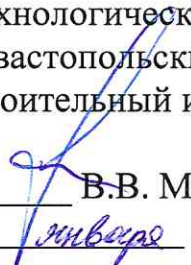
«СОГЛАСОВАНО»

Заведующий кафедрой
«Информационные технологии
и системы»


Д.В. Моисеев
« 30 » января 2026 г.

«УТВЕРЖДАЮ»

Директор института
Высшая технологическая школа
«Севастопольский
приборостроительный институт»


В.В. Мешков
« 30 » января 2026 г.

РАБОЧАЯ ПРОГРАММА ПРАКТИКИ

Производственная практика
(вид практики)

Б2.В.ДВ.01.01(П) Инженер МО

(тип практики)

09.03.04 Программная инженерия

(код и направление подготовки / специальность)

Технологии программирования. Разработка игр

(наименование профиля подготовки / специализации)

бакалавриат

(уровень высшего образования)

очная, 2026

(форма обучения, год набора)

Севастополь
2026

Рабочая программа практики составлена на основе ФГОС 09.03.04 «Программная инженерия», утвержденного приказом Министерства образования и науки Российской Федерации от 19.09.2017 №920.

Впервые утверждена и введена в действие на заседании кафедры «Информационные технологии и системы» от «30» января 2026 г., протокол №7.

Переутверждена и введена в действие с изменениями на заседании кафедры «Информационные технологии и системы» от «15» мая 2026 г., протокол №10.

Руководитель образовательной программы:

Зав. каф. «Информационные технологии и системы»,

д.т.н., доцент

(должность, ученая степень, ученое звание)



(подпись)

Д.В. Моисеев

(И.О. Фамилия)

Рабочая программа составлена:

Рабочая программа практики
составлена:

доц. каф. ИТиС

(должность, ученая степень, ученое звание)



(подпись)

Т.В. Волкова

(И.О. Фамилия)

СОДЕРЖАНИЕ

1. Цели практики	4
2. Задачи практики	4
3. Место практики в структуре ООП	4
4. Тип практики, способ и форма ее проведения	5
5. Планируемые результаты обучения при прохождении практики, соотнесенные с планируемыми результатами освоения образовательной программы.....	5
6. Структура и содержание практики.....	6
7. Формы отчетности по практике.....	7
8. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по практике.....	8
9. Учебно-методическое и информационное обеспечение практики	10
10. Материально-техническое обеспечение практики	11
11. Бланки документов на практику.....	11
ПРИЛОЖЕНИЕ А	20
ПРИЛОЖЕНИЕ Б.....	26
ПРИЛОЖЕНИЕ В	29
ПРИЛОЖЕНИЕ Г.....	35

1. Цели практики

Цель практики – закрепить и расширить теоретические и практические знания, полученные в процессе обучения.

2. Задачи практики

Основными задачами практики являются:

- углубление и закрепление полученных теоретических знаний;
- изучение технологии применения вычислительной техники в автоматизированных системах управления на предприятии;
- изучения технологии построения, конфигурирования и администрирования компьютерных сетей на предприятии;
- исследование возможности использования на предприятии автоматизированных систем обработки информации и управления для автоматизации организационно-распорядительной деятельности, технологического процесса, складского учета, а также систем автоматизированного проектирования и ведения документации;
- экономическое обоснование необходимости внедрения автоматизированных систем управления и компьютерных сетей на предприятии;
- освоение технологического оборудования, средств вычислительной техники, контрольно-измерительных приборов и другого оборудования, приборов, средств;
- автоматизация документооборота предприятия с использованием технологий WEB-программирования.

3. Место практики в структуре ООП

Элективная практика Инженер МО относится к части, формируемой участниками образовательных отношений блока 2. Практика, базируется на учебных дисциплинах:

- Алгоритмизация и программирование;
- Объектно-ориентированное программирование;
- Проектирование баз данных;
- Разработка мобильных, веб и игровых приложений.

Изучение этих дисциплин позволяет студентам, в результате успешного усвоения теоретических курсов, иметь знания, умения и компетенции к освоению программы производственной практики.

В проведении производственной практики используются также материалы, полученные в процессе прохождения учебных практик.

4. Тип практики, способ и форма ее проведения

Тип практики – производственная (Инженер МО). Практика для студентов третьего курса проводится стационарно, концентрированно на предприятиях, в научно-исследовательских учреждениях и учебных заведениях в течение 4-х недель в соответствии с учебным планом подготовки бакалавров направления 09.03.04 «Программная инженерия».

СПОСОБ – стационарная

ФОРМА –концентрированная.

5. Планируемые результаты обучения при прохождении практики, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование компетенции	Код и наименование индикатора достижения компетенции
LC-5. Способен применять и (или) проектировать различные инструменты и инженерные практики промышленной разработки систем ИИ, развертывания и сопровождения моделей машинного обучения в продуктивной среде	LC-5.1. Осуществляет выбор инструментов и инженерных практик промышленной разработки систем ИИ, развертывания и сопровождения моделей машинного обучения в продуктивной среде Уровень освоения индикатора компетенции – продвинутый. Настраивает и кастомизирует различные инструменты и инженерные практики промышленной разработки систем ИИ, развертывания и сопровождения моделей машинного обучения в рабочей версии ПО LC-5.2. Осуществляет выбор инструментов и инженерных практик по управлению данными с необходимым уровнем доступа, контроля качества, резервирования и скоростью выполнения запросов Уровень освоения индикатора компетенции – продвинутый. Настраивает инструменты виртуализации и контейнеризации — Docker, Kubernetes LC-5.0.1. (Способен проектировать аппаратное и программное обеспечение) Уровень освоения индикатора компетенции – продвинутый. Разрабатывает и настраивает программно-аппаратные комплексы
DL-2. Способен применять и (или) разрабатывать современные архитектуры генеративных глубоких сетей	DL-2.1. Применяет известные архитектуры генеративных глубоких нейронных сетей для решения прикладной задачи (генерация текста, генерация изображений по тексту, синтез речи и т.д.), при необходимости проводя дообучение на наборах данных Уровень освоения индикатора компетенции – продвинутый. Настраивает параметры генерации под конкретную задачу. Реализует техники контролируемой генерации (guidance scale, prompt engineering). Комбинирует несколько моделей в пайплайн. Проводит domain adaptation для специфических данных. Использует продвинутые методы эффективного обучения (QLoRA, DreamBooth). Оптимизирует процесс обучения (подбор lr, батчей). DL-2.2. Кастомизирует архитектуры под задачи (изменение latent space в GAN). Реализует сложные пайплайны обучения (multi stage training). Применяет продвинутые техники дообучения (LoRA, DreamBooth).

	Настраивает распределенное обучение (DDP, DeepSpeed). Оптимизирует память и скорость обучения (gradient checkpointing). Уровень освоения индикатора компетенции – продвинутый. Кастомизирует архитектуры под задачи (изменение latent space в GAN). Реализует сложные пайплайны обучения (multi stage training). Применяет продвинутое техники дообучения (LoRA, DreamBooth). Настраивает распределенное обучение (DDP, DeepSpeed). Оптимизирует память и скорость обучения (gradient checkpointing).
ML-2. Способен применять фундаментальные принципы и методы машинного обучения включая подготовку данных оценку качества моделей и работу с признаками	ML-2.1. Различает основные типы задач машинного обучения и применяет на практике принципы их решения Уровень освоения индикатора компетенции – продвинутый. Проектирует и реализует комплексные решения машинного обучения для нестандартных задач, включая разработку пайплайнов, оптимизацию моделей и интерпретацию результатов ML-2.2. Применяет методы предварительной обработки данных и работы с признаками Уровень освоения индикатора компетенции – базовый. Проводит разведочный анализ данных, умеет работать с пропущенными значениями и выбросами.
ML-4. Способен применять методы обучения без учителя для анализа структуры данных и выявления скрытых закономерностей	ML-4.1. Применяет алгоритмы кластеризации и понижения размерности для решения практических задач Уровень освоения индикатора компетенции – продвинутый. Владеет инструментами очистки данных и предварительной подготовки данных методами понижения размерности и визуализации для анализа данных. ML-4.3. Оценивает качество результатов обучения без учителя Уровень освоения индикатора компетенции – продвинутый. Применяет метрики качества кластеризации (silhouette score adjusted rand index). Комбинирует различные методы для комплексного анализа данных. Интерпретирует результаты в контексте предметной области

6. Структура и содержание практики

6.1 Структура практики

Общая трудоемкость практики составляет 6 зачетных единиц, 4 недели.

№ раздела	Наименование раздела практики	Виды учебной нагрузки и их трудоемкость, часы
		СРС
1	Подготовительный этап	4
2	Производственный этап	180
3	Подготовка отчета	32
ВСЕГО		216

6.2 Содержание практики

Подготовительный этап включает: организационное собрание, заполнение документов на практику, получение индивидуального задания.

Производственный этап включает: сбор материалов по ИЗ, обработка и анализ полученной информации, проектирование и реализацию программно-аппаратной части задания. Варианты заданий на практику представлены в приложении А. Краткие теоретические сведения приведены в приложении Б. Пример выполнения задания приведен в приложении В.

По результатам практики подготавливается отчет. Требования к составу и оформлению отчета приведены в Приложении Г.

7. Формы отчетности по практике

После окончания практики студент вместе с руководителем от кафедры обсуждает итоги производственной практики и анализирует собранные материалы.

По результатам практики формулируется тема бакалаврской работы. В ряде случаев допускается изменение темы производственной работы, т.е. формулируется новая тема, вне темы практики.

В дневнике по практике руководитель дает отзыв о работе студента, ориентируясь на его письменный отчет, доклад и отзыв руководителя от производственной организации, приведенный в дневнике.

Студент пишет отчет по практике (20 - 25 стр.), который включает в себя общие сведения о структуре предприятия, отдела или лаборатории, где проходила практика, описание постановки задачи, методы и средства её решения. К отчету прилагаются исходная конструкторская и технологическая документация.

Защита отчета о практике производится на комиссии кафедры не позднее установленного срока. Комиссия, после сообщения студента о результатах практики, вопросов и обсуждения, объявляет оценку.

Структура отчета по итогам освоения программы практики

Текст отчета практике набирается в MicrosoftWord в формате А-4: шрифт TimesNewRoman – обычный, размер 14 пт; междустрочный интервал – полуторный; левое, верхнее и нижнее поля – 2,0 см; правое поле – 1,0 см; левое поле – 3см, абзац – 1,25 см. Объем отчета должен быть 20 - 25 страниц.

Страницы отчета нумеруют арабскими цифрами, с соблюдением сквозной нумерации по всему тексту. Номер проставляется в центре нижней части листа (выравнивание от центра) без точки в конце номера. Цифровой материал должен оформляться в виде таблиц.

Исходя из указанного объема текста отчета, он должен включать следующие основные разделы:

- введение;
- основную часть;
- заключение;
- приложения (не засчитываются в объем отчета по практике).

Основные требования, предъявляемые к содержанию отчета и его структурным элементам:

Введение:

- цель, место, дата начала и продолжительность практики;
- перечень основных ознакомительных мероприятий и работ.

Основная часть

Заключение:

- необходимо оценить положительные и негативные стороны изученных мероприятий;
- показать возможности развития знаний, умений и навыков по итогам практики.

Рекомендуемый объем введения 1-1,5 страницы.

Рекомендуемый объем основной части 65 - 70% от общего количества страниц.

Рекомендуемый объем заключения 1-2 страницы.

Заключение содержит обобщение практических результатов, изложенных в основной части.

Список использованной литературы отражает источники, которые используются при изучении тем практики.

Студенты при прохождении практики обязаны в конце дня записывать выполненную работу в дневнике-графике.

8. Фонд оценочных средств для проведения промежуточной аттестации обучающихся по практике

8.1. Порядок оценивания результатов прохождения практик обучающимися;

Оценка по практике выставляется на основе защиты отчета о практике и представляет дифференцированный зачет (с оценкой). При оценке результатов практики учитывается степень самостоятельности студентов, характер и эффективность помощи, которая была оказана предприятию, и трудовая дисциплина. В таблице отражена балльно – рейтинговая система оценивания.

Таблица соответствия результатов контроля знаний по разным шкалам и критерии оценивания

Сумма баллов по 100-балльной шкале	Оценка ECTS	Критерии оценивания	Уровень компетентности	Оценка по национальной шкале	
				для экзамена, КП (КР), практики	для зачета
90 – 100	A	Индивидуальное задание выполнено полностью. При выполнении задания студент работал творчески, инициативно, самостоятельно. Отчет оформлен грамотно. Доклад по результатам практики сделан четко и ясно.	Высокий (творческий)	отлично	зачтено
82-89	B	Индивидуальное задание выполнено полностью. При выполнении задания студент работал творчески, инициативно, самостоятельно. Есть недочеты в оформлении отчета и/или в докладе.	Достаточный	хорошо	
74-81	C	Индивидуальное задание выполнено полностью. Руководитель практики влиял на результаты выполнения задания. Есть недочеты в оформлении отчета и/или в докладе.			
69-73	D	Задание выполнено частично (70%). Есть недочеты в оформлении отчета и/или в докладе.	Средний	удовлетворительно	
60-68	E	Задание выполнено частично (50%). Есть недочеты в оформлении отчета и/или в докладе.			
35-59	FX	Задание не выполнено, отчет не оформлен.	Низкий	не удовлетворительно	не зачтено
1-34	F	Задание не выполнено, отчет не оформлен. Студент не являлся на место прохождения практики.			

8.2. Типовые контрольные задания или иные материалы, необходимые для оценки результатов обучения, характеризующих этапы формирования компетенций

Типовые контрольные задания приведены в Приложении А.

9. Учебно-методическое и информационное обеспечение практики

9.1 Основная литература

1	Ларри Ульман MySQL [Электронный ресурс]/ Ларри Ульман— Электрон. текстовые данные.— М.: ДМК Пресс, 2007.— 352 с.— Режим доступа: http://www.iprbookshop.ru/7848 .— ЭБС «IPRbooks», по паролю
2	Нолан Хестер Как создать превосходный сайт в Microsoft Expression Web 2 и CSS [Электронный ресурс]/ Нолан Хестер— Электрон. текстовые данные.— М.: ДМК Пресс, 2009.— 256 с.— Режим доступа: http://www.iprbookshop.ru/7951 .— ЭБС «IPRbooks», по паролю
3	Савельев А.О. HTML 5. Основы клиентской разработки [Электронный ресурс]/ Савельев А.О., Алексеев А.А.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2012.— 166 с.— Режим доступа: http://www.iprbookshop.ru/16680 .— ЭБС «IPRbooks», по паролю
4	Магазанник В.Д. Человеко-компьютерное взаимодействие [Электронный ресурс]: учебное пособие/ Магазанник В.Д.— Электрон. текстовые данные.— М.: Логос, Университетская книга, 2011.— 256 с.— Режим доступа: http://www.iprbookshop.ru/9113 .— ЭБС «IPRbooks», по паролю

9.2 Дополнительная литература

1	Савельева Н.В. Основы программирования на PHP [Электронный ресурс]: курс лекций. Учебное пособие для студентов вузов, обучающихся по специальностям в области информационных технологий/ Савельева Н.В.— Электрон. текстовые данные.— М.: Интернет-Университет Информационных Технологий (ИНТУИТ), 2005.— 264 с.— Режим доступа: http://www.iprbookshop.ru/22429 .— ЭБС «IPRbooks», по паролю
2	Бенкен, Е. С. PHP, MySQL, XML: программирование для Интернета / Е.С.Бенкен. — СПб.: БХВ- Петербург, 2007. — 314 с. - ISBN 5-9775-0039-4.
3	Кузнецов, М. В. PHP. Практика создания Web-сайтов / М. В. Кузнецов, И. В. Симдянов. — 2-е изд., перераб. и доп. — СПб.: БХВ-Петербург, 2008. — 1251 с.: ил. + CD-ROM — (Профессиональное программирование). - ISBN 978-5-9775-0203-0.

9.3 Методические указания по практике

Методические указания по прохождению производственной практики для студентов очной формы обучения по направлению 09.03.04 – «Программная инженерия» доступны в дистанционном курсе <https://do.sevsu.ru/course/view.php?id=2049>.

9.4 Информационные технологии

Перечень информационных технологий, используемых при проведении практики:

1. Пакет MS Office
2. Системы управления контентом (Joomla 3.0, WordPress и т.п.)
3. Специализированные Web-редакторы (DreamWeaver и т.п.)

4. Комплект Web-разработчика (Denwer, XAMPP и т.п.)

10. Материально-техническое обеспечение практики

Материально-техническая база предприятия, на котором проводится технологическая практика, должна соответствовать действующим санитарным и противопожарным правилам и нормам и может включать в себя:

1. Компьютеры, частично или полностью оснащенные программным обеспечением, приведенным в пункте 9.4 настоящей программы (или аналогами).
2. Компьютерную сеть, с использованием современного сетевого оборудования (сервера, свитчи, роутеры, маршрутизаторы и т.д.).
3. Неограниченный доступ в интернет с возможностью использования статических IP адресов.
4. Другое оборудование необходимое для проведения производственной практики.

11. Бланки документов на практику

Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

ДНЕВНИК ПРАКТИКИ

(вид и тип практики)

обучающегося

(фамилия, имя, отчество)

Институт

Кафедра

Уровень образования

Направление подготовки
(специальность)

Профиль (специализация)

_____ курс, группа _____

Обучающийся _____
(фамилия, имя, отчество)

Прибыл на предприятие (в организацию, учреждение)

М.П. “ ____ ” _____ 20 ____ г.

(подпись)

(должность, фамилия и инициалы ответственного лица)

Отметки о проведении инструктажа

Наименование инструктажа	Дата проведени я инструкта жа	ФИО, должность инструктирующего	Подпись	
			Инструкт и- рующего	Инструкт и- руемого
Инструктаж по технике безопасности				
Инструктаж по охране труда				
Инструктаж по пожарной безопасности				
Инструктаж по правилам внутреннего распорядка				

Убыл из предприятия (организации, учреждения)

М.П. “ ____ ” _____ 20 ____ г.

(подпись)

(должность, фамилия и инициалы ответственного лица)

(подпись)

(инициалы и фамилия)

(подпись)

(инициалы и фамилия)

« _____ » 20 ____ г.

Рабочие записи во время практики

[illegible]

Руководитель практики от профильной организации

(инициалы и фамилия)

« _____ » 20 ____ г.

Сведения об уровне освоения обучающимися компетенций

№	Формируемые компетенции	Руководитель от профильной организации	Руководитель от Университета
1.	ОК- (Наименование компетенций заполняет руководитель практики от института, в соответствии с учебным планом)	освоена / освоена частично/ не освоена	освоена / освоена частично/ не освоена
2.	ОК-	освоена / освоена частично/ не освоена	освоена / освоена частично/ не освоена
3.	ОПК -	освоена / освоена частично/ не освоена	освоена / освоена частично/ не освоена
4.	ПК -	освоена / освоена частично/ не освоена	освоена / освоена частично/ не освоена
5.	ПК -	освоена / освоена частично/ не освоена	освоена / освоена частично/ не освоена
6.	...		

Руководитель практики от Университета

(подпись)

(инициалы и фамилия)

Руководитель практики от профильной организации

(подпись)

(инициалы и фамилия)

« ____ » _____ 20 ____ г.

Оценка результатов прохождения практики обучающимся

(заполняется руководителем практики от Университета)

(Фамилия, имя, отчество обучающегося)

Дата сдачи отчета « » 20 г.

Оценка: _____

Руководитель практики от Университета

(подпись)

(инициалы и фамилия)

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«СЕВАСТОПОЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

Институт _____

Кафедра _____

№	Дата поступления на кафедру	Подпись отв. за регистрацию	Подпись преподавателя

ОТЧЕТ

о _____ (_____) практике
(вид практики) (тип практики)

в _____
(наименование организации)

Выполнил _____
(Фамилия И.О. обучающегося)

_____ (шифр группы)

Направление / специальность _____

_____ (код, наименование)

Руководитель практики от Университета

_____ (должность)

_____ (Фамилия И.О. руководителя)

Севастополь

20__ г.

ПРИЛОЖЕНИЕ А

Варианты индивидуальных заданий

Номер варианта определяется преподавателем.

Перед выполнением индивидуального задания рекомендуется выполнить пример из приложения В.

Вариант 1

Дано: список сотрудников и занимаемых ими должностей, список должностей и окладов. Список должностей и окладов разместить в базе данных.

Сформировать список сотрудников, отправляемых на переподготовку. Командировочные расходы установить в размере оклада умноженного на сумму коэф. К1 и К2. Значения коэффициентов располагаются в отдельных полях. В итоговой строке предусмотреть расчет суммарных затрат.

Предусмотреть кнопку «Обновить оклады».

Вариант 2

Рецепт

Дано: список лекарств и цен, список лекарств и рекомендуемых доз, поле, в котором указан процент льготы

В итоговой строке произвести расчет общей стоимости.

Предусмотреть кнопку «Обновить рекомендуемые дозы»

Вариант 3

Дано: Список оборудования и их кодов, список оборудования с первоначальной стоимостью, список кодов ОС и нормативным сроком службы.

Требуется сформировать акт на поступление/передачу ОС, включающий следующие поля

Наименование основного средства	Код основного средства	Первоначальная стоимость	Дата поступления ОС	Нормативный срок службы	Норма амортизации годовая	Норма амортизации за месяц

В таблице поле «Норма амортизации годовая» и поле «Норма амортизации за месяц» рассчитываются по формулам:

Показатель *Норма амортизации годовая*:

= 1/[Нормативный срок службы].

Показатель *Норма амортизации за месяц*:

= 1/[Нормативный срок службы] / 12.

Список кодов ОС и нормативных сроков службы разместить в базе данных.

Предусмотреть кнопку «Обновить нормативный срок службы».

В итоговой строке предусмотреть расчет суммарной стоимости переданных ОС.

Вариант 4

Дано: Список оборудования и их кодов, список оборудования с первоначальной стоимостью, список кодов ОС и нормативным сроком службы

Требуется сформировать акт на выбытие ОС, включающий следующие поля

Наименование основного средства	Код основного средства	Первоначальная стоимость	Дата начала эксплуатации	Дата выбытия	Код вида выбытия	Нормативный срок службы	Накопленная амортизация	Остаточная стоимость

Дополнительно есть списки видов выбытия ОС, поля с датами поступления и выбытия.
Расчет показателей:

1) Показатель *Накопленная амортизация*:

= ([Дата выбытия] - [Дата начала эксплуатации]) / 30 * [Норма амортизации за месяц] * [Первоначальная стоимость].

2) Показатель *Остаточная стоимость*:

= [Первоначальная стоимость] - [Накопленная амортизация].

В итоговой строке предусмотреть расчет суммарной остаточной стоимости выбывших ОС.

Вариант 5

Рецепт

Дано: список лекарств и цен, список лекарств и рекомендуемых доз, поле, в котором указан процент льготы

В итоговой строке произвести расчет общей стоимости.

Предусмотреть кнопку «Обновить цены»

Вариант 6

Список процедур

Дано: список процедур со стоимостью за 10 минут, список длительностей процедур, поле, в котором указано предельное суммарное время, отводимое на процедуры.

В итоговой строке произвести расчет общей длительности. Она не может превышать суммарного предельного времени

Предусмотреть кнопку «Обновить список длительностей процедур»

Вариант 7

Список процедур

Дано: список процедур со стоимостью за 10 минут, список длительностей процедур.

В итоговой строке произвести расчет общей стоимости.

Предусмотреть кнопку «Обновить цены на процедуры»

Вариант 8

Продажа недвижимости

Дано: Список объектов недвижимости и цен, описание объекта недвижимости (кол-во комнат, метраж и т.п.), поле, в котором указан процент комиссии

В итоговой строке произвести расчет общей стоимости

Предусмотреть кнопку «Обновить цены на объекты недвижимости»

Вариант 9

Тренажерный зал

Дано: поле, в котором указан занимающийся, список тренажеров со стоимостью абонирования за 1 час., поле времени абонирования,
В итоговых строках произвести расчет процента времени использования тренажеров.
Предусмотреть кнопку «Обновить стоимость абонирования тренажеров»

Вариант 10

Туристический бизнес

Дано: список туров, цен и скидок, поле с ФИО покупателя

В итоговой строке произвести расчет средней цены за тур, сумму, потраченную на скидки

Предусмотреть кнопку «Обновить цены»

Вариант 11

Туристический бизнес

Дано: список туров, цен и скидок, поле с ФИО покупателя

В итоговой строке произвести расчет максимальной цены за тур, сумму, потраченную на скидки

Предусмотреть кнопку «Обновить скидки»

Вариант 12

Учебный план. Дисциплины по выбору

Дано: список дисциплин с количеством часов, поле, указывающее семестр

В итоговой строке произвести расчет итогового количества часов

Предусмотреть кнопку “Обновить список часов дисциплин”

Суммарное число часов не должно превышать предела, заданного в соответствующем поле

Вариант 13

Учебный план. Дисциплины по выбору

Дано: список дисциплин с количеством часов, поле, указывающее семестр

В итоговой строке произвести расчет итогового количества часов

Предусмотреть кнопку “Обновить список часов дисциплин”

В таблицу не может быть включено предметов, читаемых в одном семестре, больше значения, заданного в соответствующем поле.

Вариант 14

Список поставщиков

Дано: Список поставщиков товаров и цен.

В результирующую таблицу включить тех, кто поставяет нужный товар, заданный в некотором поле и не превышающем значения цены, указанного в поле.

Предусмотреть кнопку обновления цены для некоторого поставщика

В итоговую строку включить информацию о поставщике товара с минимальной ценой

Вариант 15

Список поставщиков

Дано: Список поставщиков товаров и цен.

В результирующую таблицу включить тех, кто поставяет нужный товар, заданный в некотором поле и не превышающем значения цены, указанного в поле.

Предусмотреть кнопку обновления цены для некоторого поставщика

В итоговую строку включить информацию о средней цене на заданный товар

Вариант 16

Меню

Дано: списки первых блюд и цен, вторых блюд и цен, десертов и цен, напитков и цен
 В итоговой строке произвести расчет итоговой суммы заказа
 Предусмотреть кнопки обновить цены на блюда.

Вариант 17

Бюро по трудоустройству и профориентации

Дано: список вакансий и зарплат. Подобрать вариант по профессии и интервалу зарплат, заданных в полях №1 и №2 соответственно.

В итоговой строке привести вариант с максимальной зарплатой.

Предусмотреть кнопку обновления зарплат списка вакансий.

Вариант 18

Бюро по трудоустройству и профориентации

Дано: список вакансий и зарплат. Подобрать вариант по профессии и интервалу зарплат, заданных в полях №1 и №2 соответственно.

В итоговой строке привести все подходящие варианты.

Предусмотреть кнопку обновления зарплат списка вакансий.

Вариант 19

Гостиница

Дано: список номеров, их типа, стоимости за сутки, поле для отбора номера по типу

В итоговой строке привести вариант со средней ценой.

Предусмотреть кнопку обновления стоимости номеров за сутки.

Вариант 20

Учет рабочего времени

Дано: список тарифных ставок(код разряда; часовая тарифная ставка;коэффициент доплаты за сверхурочную работу;), список табель (Ф.И.О. работающего; код разряда; отработанное время, всего)

Сформировать таблицу учета зарплаты

Ф.И.О. рабочего	Часовая тарифная ставка	Отработа нное время (всего)	Основная заработн ая плата
--------------------	-------------------------------	--------------------------------------	----------------------------------

Основную зарплату рассчитать по формуле [Часовая тарифная ставка]*[Фактически отработанное время].

В итоговой строке рассчитать суммарное отработанное время и суммарную зарплату.

Предусмотреть кнопку обновления списка тарифных ставок

Вариант 21

Учет рабочего времени

Дано: список тарифных ставок(код разряда; часовая тарифная ставка;коэффициент доплаты за сверхурочную работу;), список табель (Ф.И.О. работающего; код разряда; отработанное время, всего)

Сформировать таблицу учета дополнительной зарплаты

Ф.И. О. работч его	Часов ая тариф ная ставка	Отрабо танн ое время (всего)	Допла та за сверху рочно отрабо танное время
-----------------------------	---------------------------------------	--	--

Алгоритм вычисления показателя: *Доплата за сверхурочно отработанное время:*

$$= ([\text{Отработанное время (всего)}] - [\text{Фонд рабочего времени отчетного месяца}]) * [\text{часовая тарифная ставка}] * [\text{коэффициент доплаты}].$$

Фонд рабочего времени отчетного месяца выбирать из отдельного поля (Обычное значение 155-165 часов).

В итоговой строке рассчитать суммарное отработанное время и суммарную доплату.

Предусмотреть кнопку обновления списка «коэффициент доплаты за сверхурочную работу».

ПРИЛОЖЕНИЕ Б

Основные теоретические сведения

HTML DOM (Document Object Model) – представляет собой стандарт консорциума W3C для программного доступа к документам HTML или XML. Фактически это платформи- и языково-нейтральный интерфейс, позволяющий программам и сценариям динамически обращаться и обновлять содержимое, структуру и стиль HTML-документа.

Согласно модели DOM:

- Весь документ представляется узлом документа;
- Каждый HTML тэг является узлом элемента;
- Текст внутри HTML элементов представляется текстовыми узлами;
- Каждому HTML атрибуту соответствует узел атрибута;
- Комментарии являются узлами комментариев.

Например:

```
<html>
<head>
  <title>HTML документ</title>
</head>
<body>
  <h1>Заголовок </h1>
  <p>Просто текст</p>
</body>
</html>
```

В этом примере корневым узлом является тэг `<html>`. Все остальные узлы содержатся внутри `<html>`. У этого узла имеется два дочерних узла: `<head>` и `<body>`. Узел `<head>` содержит узел `<title>`, а узел `<body>` содержит узлы `<h1>` и `<p>`.

В рамках DOM модели HTML можно рассматривать как множество узловых объектов. Доступ к ним осуществляется с помощью JavaScript или других языков программирования. Программный интерфейс DOM включает в себя набор стандартных свойств и методов.

К типичным свойствам объектов DOM относятся следующие:

- `x.innerHTML` – внутреннее текстовое значение HTML элемента `x` ;
- `x.nodeName` – имя `x` ;
- `x.nodeValue` – значение `x` ;
- `x.parentNode` – родительский узел для `x` ;
- `x.childNodes` – дочерний узел для `x` ;
- `x.attributes` – узлы атрибутов `x`.

Узловой объект, соответствующий HTML элементу поддерживает следующие методы:

- `x.getElementById(id)` – получить элемент с указанным `id` ;
- `x.getElementsByTagName(name)` – получить все элементы с указанным именем тэга (`name`) ;
- `x.appendChild(node)` – вставить дочерний узел для `x` ;
- `x.removeChild(node)` – удалить дочерний узел для `x`.

Например:

Для получения текста из элемента `<p>` со значением атрибута `id "demo"` в HTML документе можно использовать следующий код:

```
txt = document.getElementById("demo").innerHTML
```

Тот же самый результат может быть получен по-другому:

```
txt=document.getElementById("demo").childNodes[0].nodeValue
```

В рамках DOM возможны 3 способа доступа к узлам:

1. С помощью метода `getElementById(ID)`. При этом возвращается элемент с указанным ID.
2. С помощью метода `getElementsByTagName(name)`. При этом возвращаются все узлы с указанным именем тэга (в виде индексированного списка). Первый элемент в списке имеет нулевой индекс.
3. Путем перемещения по дереву с использованием отношений между узлами.

Для определения длины списка узлов используется свойство `length`. Например:

```
x = document.getElementsByTagName("p");
for (i = 0; i < x.length; i++)
{
    document.write(x[i].innerHTML);
    document.write("<br/>");
}
```

В данном примере внутрь HTML документа вставляется в виде списка текстовое содержимое всех элементов соответствующих тэгу `<p>`.

Для навигации по дереву в ближайших окрестностях текущего узла можно использовать следующие свойства:

- `parentNode` ;
- `firstChild` ;
- `lastChild`.

Ниже приводится пример, в котором пользователь имеет возможность выбрать цвет текста с помощью диалогового элемента

```
<html>
<body>
// здесь будет отображаться текст
<div id="c" style="color:blue">Вы выбрали цвет текста: синий</div>

<script language="JavaScript">
// пользователь выбирает цвет текста
var tcolor = prompt("Выберите цвет текста: red, blue, green, yellow, black","black");
// задается текст
document.getElementById("c").innerHTML = "Вы выбрали цвет текста: " + tcolor;
// задается цвет текста
document.getElementById("c").style.color = tcolor;

</script>
</body>
</html>
```

Методы, применимые к table:

- createCaption() - создать элемент caption
- deleteCaption() - удалить элемент caption
- createTHead() - создать элемент thead
- deleteTHead() - удалить элемент thead
- createTFoot() - создать элемент tfoot
- deleteTFoot() - удалить элемент tfoot

Методы, применимые к table, thead, tbody и tfoot:

- insertRow(index) - добавить строку в указанное место
- deleteRow(index) - удалить строку с указанным номером
- moveRow(fromIndex, toIndex) - переместить строку

Методы, применимые к tr:

- insertCell(index) - добавить ячейку в указанное место
- deleteCell(index) - удалить ячейку с указанным номером

Методы, имена которых начинаются на create и insert возвращают ссылку на созданный объект. Методы insertRow(index) и insertCell(index) позволяют в качестве номера передать число -1, и в этом случае элемент будет вставлен в конец. Все эти методы можно вызвать в коде сценария на Javascript.

Также полезно знать о некоторых свойствах, предоставляемых объектной моделью таблицы:

Свойства объекта table:

- tHead - доступ к свойствам заголовка;
- tBodies[] - доступ к коллекции фрагментов строк;
- tFoot - доступ к свойствам заголовка;

rows[] - также применимо к thead, tbody и tfoot.

Свойство объекта tr:

cells[] - доступ к свойствам ячейки.

ПРИЛОЖЕНИЕ В

Пример выполнения задания

Пусть существует сайт, предлагающий сформировать заказ на покупку и доставку продуктов. Для этого используется форма заказа. Рассмотрим процесс формирования клиентом заказа из набора продуктов (рис.1). При выборе клиентом продукта, указании количества единиц и нажатии кнопки "Добавить в корзину" к таблице заказа добавляется строка.

Кнопка "Удалить" позволяет удалять пункты из заказа по одному или группой. При всех манипуляциях подсчитывается число единиц заказанных товаров и их суммарная стоимость.

Продукты: Количество:

Ваш заказ

☐	Продукт	Цена	Количество	Сумма	
	Итого:		0	0.00	Удалить отмеченные

Рис. 1. Начальный вид формы заказа при открытии страницы

Решение поставленной задачи выполним в два этапа: На первом создадим форму средствами HTML и CSS; на втором подготовим скрипты Javascript для динамического создания таблицы заказа. Назовем файл «Order.html».

Часть 1. Создание разметки

Рассмотрим действия по созданию формы на рис.1 пошагово.

1. Создание выпадающего списка (элемент select).

В теле нового документа создайте элемент div, в котором будет размещена вся разметка данного примера.

Первый элемент - select - должен содержать несколько групп (optgroup) по несколько пунктов (option). Каждый пункт содержит наименование какого-либо продукта, а его атрибут value - цену (произвольное в данном случае число).

В развёрнутом виде список должен выглядеть так:

Картошка

Овощи

Картошка

Морковь

Лук

Фрукты

Яблоки

Бананы

Напитки

Чай

Кофе

Назначьте списку атрибут `id="lstProducts"`.

2. Создание остальной разметки

Следом за выпадающим списком создайте два элемента ввода - `input type="text"` и `input type="button"` (примеры приведены в лекции). Текстовому элементу ввода назначьте атрибут `id="txtQty"`, а кнопке - `onclick="AddToCart()"`.

Последний элемент разметки - таблица с заголовком (элемент `caption`), верхним и нижним колонтитулом (элементы `thead` и `tfoot`) и телом (элемент `tbody`). Описания и примеры применения этих элементов также см. в приложении. Таблице назначьте атрибут `id="tblOrder"`.

Раздел `tbody` оставьте пустым (он будет формироваться динамически), а в колонтитулах, как видно на рисунке, следует расположить по одному элементу управления - `checkbox` и `button`. Им сразу задайте обработчики события `click` следующим образом:

```
<input type="checkbox" onclick="ToggleCheck(this)" />
<input type="button" value="Удалить отмеченные" onclick="RemoveSelected()" />
```

Часть 2. Создание сценария, манипулирующего таблицей

1. Добавление пунктов заказа

Создайте скрипт в конце тела документа, и сразу определите обработчики, назначенные элементам управления в предыдущем задании - пока пусть это будут функции с пустым телом.

В начало скрипта поместите следующие команды:

```
var tbl = document.getElementById('tblOrder');
var oList = document.getElementById('lstProducts');
```

Ссылки на элементы `'tblOrder'` (таблица заказа) и `lstProducts` (список выбора продукта) будут часто использоваться в коде сценария, поэтому целесообразно определить их единожды.

При нажатии кнопки "Добавить в корзину" в таблицу должны добавляться строки, соответствующие пунктам заказа:

Продукты: Количество:

Ваш заказ

☐	Продукт	Цена	Количество	Сумма	
☐	Картошка	13.95	2	27.90	Удалить
☐	Яблоки	69.90	3	209.70	Удалить
Итого:			5	237.60	Удалить отмеченные

Функция-обработчик этой кнопки была названа `AddToCart`. Изучите её код и добавьте его в скрипт.

```
/* Добавление пунктов заказа */
function AddToCart() {
    /* Определяем значение, введённое в текстовое поле */
    var qty = document.getElementById('txtQty').value;
    /* Проверка: распознаётся ли значение как число? Если нет, считаем
    единицей */
    if (parseFloat(qty) != qty)
        qty = 1;
    /* Вставляем строку в тело таблицы */
    var oRow = tbl.tBodies[0].insertRow(-1);
    /* В добавленную строку вставляем, во-первых, checkbox */
```

```

oRow.insertCell(-1).innerHTML = '<input type="checkbox">';
/* во-вторых, текст, взятый из списка выбора продуктов */
oRow.insertCell(-1).innerHTML = oList.options[oList.selectedIndex].text;
/* в-третьих, цена выбранного продукта */
oRow.insertCell(-1).innerHTML = oList.value;
/* далее, количество, указанное в текстовом поле */
...
/* затем стоимость пункта заказа */
...
/* и, наконец, кнопку "Удалить" */
...
/* По окончании вставки строки необходимо пересчитать сумму заказа */
//Calculate();
}

```

Код в тех строках, где оставлено многоточие, напишите самостоятельно, опираясь на комментарии. Вызов функции Calculate() оставьте пока закомментированным - эту функцию мы добавим позже, а сначала следует добиться верной работы AddToCart().

2. Функция Calculate.

Создайте функцию Calculate() и уберите комментарий с её вызова в функции AddToCart.

```

function Calculate() {
  /* Счётчики для количества единиц товара и общей стоимости */
  var qty = 0, amount = 0;
  /* Цикл по всем строкам в теле таблицы */
  for (var i = 0, n = tbl.tBodies[0].rows.length; i < n; i++) {
    /* Увеличиваем qty на значение в 3 столбце текущей строки */
    qty += parseFloat(tbl.tBodies[0].rows[i].cells[3].innerHTML);
    /* Увеличиваем amount на значение в 4 столбце текущей строки */
    amount += parseFloat(tbl.tBodies[0].rows[i].cells[4].innerHTML);
  }
  /* Записываем qty в 3 столбец нижнего колонтитула */
  ...
  /* Записываем amount в 4 столбец нижнего колонтитула */
  ...
}

```

Код в тех строках, где оставлено многоточие, напишите самостоятельно, опираясь на комментарии.

3. Функция RemoveProduct(elem).

Кнопка "Удалить", динамически вставляемая в правую ячейку каждой строки заказа, должна выполнять свою работу - следовательно, необходимо назначить ей обработчик (назовём его RemoveProduct) и написать его код. Этот обработчик имеет существенное отличие от предыдущих, которое заключается в том, что смысл его работы зависит от контекста вызова: удалять нужно именно ту строку, в которой кнопка расположена. Таким образом, обработчик должен принимать параметр, определяющий контекст. Это можно сделать различными путями, и здесь мы предлагаем наиболее типичный. Если назначить кнопке обработчик следующим образом: onclick="RemoveProduct(this)", то функция RemoveProduct получит в качестве параметра ссылку на тот объект, который принял событие (в данном случае - щелчок мыши). Очевидно, обладая ссылкой на элемент в таблице, можно каким-то образом определить номер (index) строки таблицы, в которой он находится, и применить метод таблицы deleteRow(index). Отношение между строкой и кнопкой "Удалить" такое: строка является контейнером ячейки, которая является контейнером кнопки. Ссылку на контейнер элемента можно получить при помощи свойства parentNode этого элемента. С учётом этих соображений код рассматриваемой функции примет следующий вид:

```

function RemoveProduct(elem) {
  tbl.deleteRow(elem.parentNode.parentNode.rowIndex);
  Calculate();
}

```

```
}
```

Вставьте эту функцию в скрипт (и не забудьте правильно описать её вызов в динамическом определении кнопки - см. функцию AddToCart). Сохраните документ, обновите страницу в браузере и проверьте добавленную функциональность.

4. Функция RemoveSelected.

Код этой функции представлен ниже, и на нём следует остановиться подробнее, поскольку работа по удалению элементов из множества имеет свои тонкости.

```
function RemoveSelected() {
    /* находим все элементы input в теле таблицы */
    var checks = tbl.tBodies[0].getElementsByTagName('input');
    var i = 0;
    /* начинаем перебор элементов в цикле */
    while (i < checks.length) {
        /* рассматриваем элемент лишь в том случае, если это checkbox и он
отмечен */
        if (checks[i].type == 'checkbox' && checks[i].checked)
            /* вызываем функцию, которая удалит строку с пунктом заказа -
передаём ей ссылку на checkbox */
            RemoveProduct(checks[i]);
        else
            /* счётчик увеличиваем лишь в том случае, если удаление не было
сделано */
            i++;
    }
}
```

Множество ссылок на все элементы ввода (input) внутри тела таблицы (tbody) легко получить при помощи метода getElementByTagName. Этот метод принимает в качестве параметра имя тэга и выдаёт массив ссылок. Важно отметить, что ссылками являются не только элементы массива, но и сам массив - *ссылочный*, т.е. все изменения в документе автоматически отражаются в этом массиве. Таким образом, удалив строку из таблицы (и тем самым удалив из неё два элемента input - checkbox и button), мы тем самым уменьшим число элементов в массиве checks на 2. Это следует принимать во внимание при циклической обработке массива.

Номер рассматриваемого в цикле элемента массива обозначен в данном примере переменной i, и отсчёт, как обычно, начинается с нуля. Чтобы правильно сформулировать оператор увеличения этого i, следует задаться вопросом: "Рассмотрев элемент с номером i, элемент с каким номером следует рассматривать далее?" Ответ таков: "Если удаление не было выполнено, то i+1 ; иначе - вновь i (теперь это номер следующего из оставшихся элементов)". Поэтому цикл записан именно таким образом (а не с использованием оператора for).

6. Функция ToggleCheck.

Последняя функция данного сценария создаёт дополнительное удобство: можно отметить или сбросить сразу все галочки в пунктах заказа, щёлкнув по галочке (элементу checkbox) в заголовке таблицы.

Напишите эту функцию самостоятельно - все необходимые для этого приёмы уже рассмотрены (например, функция RemoveSelected()).

Окончательный вид работающей страницы показан на [рис.2](#).

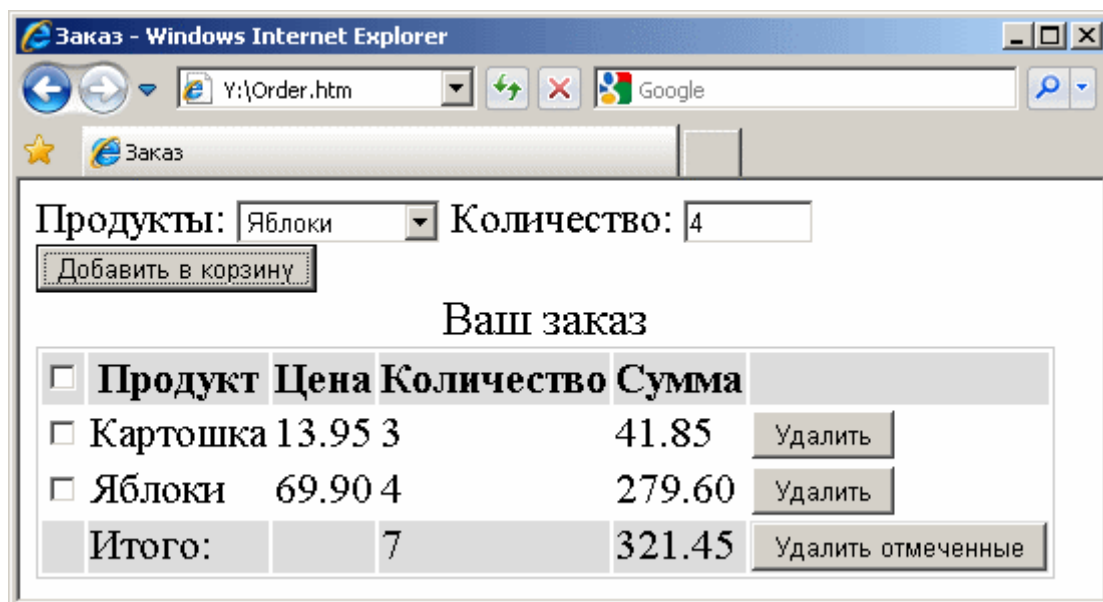


Рис. 2. Действующая страница заказа

Часть 3. Моделирование процесса взаимодействия с базой данных сервера.

Усложним задачу. Пусть необходимо получать от сервера текущие значения цен на данный момент времени. Полноценный диалог с сервером требует значительных работ по программированию, поэтому ограничимся его моделированием.

Пусть актуальные цены на товары можно ввести в форме на другой странице сайта, которую назовем «База_данных». Тогда на форме заказа можно сделать обращение к этой странице. Для этого создадим кнопку и при ее нажатии вызовем, например функцию

```
function Get_Prices(){
    self.location="База_данных.html";
}
```

На странице "База_данных.html" создадим модель базы данных.

```
<html>
<head>
<script type="text/javascript">
<!--
function newPage()
{
    value1="" + document.form1.CV.value;
    value2="" + document.form1.CS.value;
    value3="" + document.form1.CM.value;
    value4="" + document.form1.CO.value;
    self.location="Order.html?" + value1 + "=" + value2 + "=" + value3 + "=" + value4;
}
//-->
</script>
</head>
<body >
<form name="form1">
```

```

Volvo&nbsp;<input type="text" name="CV" value="120"><br>
SAAB&nbsp;&nbsp;<input type="text" name="CS" value="130"><br>
Mercedes&nbsp;<input type="text" name="CM" value="150"><br>
Opel&nbsp;<input type="text" name="CO" value="125"><br>

<input type="button" name="button" value="Подтвердить данные" onClick="newPage()">
</form>
</body>
</html>

```

Возврат в форму заказа (файл Order.html) осуществляется с помощью метода Get.

После возврата в форму заказа полученные данные следует обработать, например, следующим скриптом:

```

<script type="text/javascript">
    /* Обработка новых цен */
    userName = ""+ document.location;
    if (userName.indexOf("?")>0) {
        n1 = userName.indexOf("=")+1;
        userName= userName.substring(n1); /* Обрезаем по первый знак равно */

        n1= userName.indexOf("=")+1;
        CV=userName.substr(0,n1-1);

        userName= userName.substring(n1);
        n1= userName.indexOf("=")+1;
        CA=userName.substr(0,n1-1);

        userName= userName.substring(n1);
        n1= userName.indexOf("=")+1;
        CM=userName.substr(0,n1-1);

        CO= userName.substring(n1);

        var mylist=document.getElementById("lstProducts");
        mylist.options[0].value=CV;
        mylist.options[1].value=CA;
        mylist.options[2].value=CM;
        mylist.options[3].value=CO;
    }
</script>

```

ВНИМАНИЕ! Этот скрипт должен быть расположен в конце файла Order.html.

ПРИЛОЖЕНИЕ Г

Структура отчета о практике и требования к оформлению отчета

Структурными элементами отчета о практике являются:

- титульный лист;
- содержание;
- введение;
- основная часть;
- заключение;
- список литературы;
- приложения (должностная инструкция, тексты программ и др.).

Основная часть должна включать следующие разделы:

- 1) Описание должностной инструкции сотрудника, в должности которого проходит практика;
- 2) Постановку задачи на проектирование программного блока в соответствии с вариантом;
- 3) Техническое задание на проектирование программного блока в соответствии с вариантом (ГОСТ 19.201-78 Техническое задание. Требования к содержанию и оформлению);
- 4) Описание программного блока следующими программными документами:
 - описание программы (ГОСТ 19.402-78 Описание программы. Требования к содержанию и оформлению);
 - руководство программиста (ГОСТ 19.504-79 Руководство программиста. Требования к содержанию и оформлению);
 - программа и методика испытаний (ГОСТ 19.301-79 Программа и методика испытаний. Требования к содержанию и оформлению).

Требования к оформлению отчета о практике

1. Оформление отчета о практике должно соответствовать требованиям к текстовым учебным документам соответствующих ГОСТов.

2. Текстовая часть отчета о практике выполняется с использованием печатающих и графических устройств на одной стороне листа белой бумаги формата А4 с параметрами: междустрочный интервал – 1,5; кегль – 14; шрифт – TimesNewRoman, обычный; цвет шрифта – черный; поля, не менее:

верхнее – 20 мм; левое – 30 мм;
нижнее – 20 мм; правое – 15 мм.

3. Иллюстрационно-графический материал в зависимости от специфики программы может включать: чертежи, схемы (технологические, алгоритмов и

др.), плакаты, диаграммы, макеты, фотографии, аудио- и видеоматериалы, натурные образцы и др.

4. Отчет должен быть переплетен доступным способом.