

федеральное государственное бюджетное образовательное учреждение высшего образования "Приволжский исследовательский медицинский университет"
Министерства здравоохранения Российской Федерации



УТВЕРЖДАЮ
Проректор по учебной
и воспитательной работе
Богомолова Е.С.

2023 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине **Языки программирования**

направление подготовки **09.04.02 Информационные системы и технологии**

профиль **Информационные системы и технологии в здравоохранении**

Квалификация выпускника:
Магистр

Форма обучения:
очно-заочная

Нижний Новгород
2023

Фонд оценочных средств по дисциплине «Языки программирования» предназначен для контроля знаний по программе магистратуры по направлению подготовки 09.04.02 Информационные системы и технологии, профилю «Информационные системы и технологии в здравоохранении».

1. Паспорт фонда оценочных средств по дисциплине «Языки программирования»

Компетенция	Результаты обучения и индикаторы достижения компетенций	Виды занятий	Оценочные средства
ОПК-2	<p>способен разрабатывать оригинальные алгоритмы и программные средства, в том числе с использованием современных интеллектуальных технологий, для решения профессиональных задач</p> <p>Уметь: ИД-3_{ОПК-2.3} разрабатывать алгоритмы и программные средства для решения профессиональных задач.</p>	Лекции, Практические и семинарские занятия, Самостоятельная работа,	Контрольная работа, Собеседование, Тестирование
ОПК-5	<p>способен разрабатывать и модернизировать программное и аппаратное обеспечение информационных и автоматизированных систем</p> <p>Знать: ИД-1_{ОПК-5.1} принципы разработки сетевых приложений и основы облачных вычислений; Уметь: ИД-3_{ОПК-5.3} модифицировать исходный код разработанных программных продуктов для решения определенной задачи (устройствам хранения данных, приложениям и сервисам); ИД-4_{ОПК-5.4} предоставлять сетевой доступ к фонду вычислительных ресурсов; Владеть: ИД-7_{ОПК-5.7} навыками разработки и модернизации программного и аппаратного обеспечения информационных и автоматизированных систем; ИД-8_{ОПК-5.8} использовать свойства эластичных вычислений облачных услуг.</p>	Лекции, Практические и семинарские занятия, Самостоятельная работа,	Контрольная работа, Собеседование, Тестирование
ПК-3	<p>способен выполнять планирование, мониторинг и управление проектами с применением современных методов и инструментальных средств</p> <p>Знать: ИД-2_{ПК-3.2} принципы построения баз данных, синтаксис SQL и язык разработки PHP. Уметь: ИД-10_{ПК-3.10} составлять техническую документацию к разрабатываемому продукту. Владеть: ИД-17_{ПК-3.17} навыками коллективной</p>	Лекции, Практические и семинарские занятия, Самостоятельная работа,	Контрольная работа, Собеседование, Тестирование

<i>Компетенция</i>	<i>Результаты обучения и индикаторы достижения компетенций</i>	<i>Виды занятий</i>	<i>Оценочные средства</i>
	разработки сложных программных продуктов.		

Текущий контроль по дисциплине «Языки программирования» осуществляется в течение всего срока освоения данной дисциплины. Выбор оценочного средства для проведения текущего контроля на усмотрение преподавателя.

Промежуточная аттестация обучающихся по дисциплине «Языки программирования» проводится по итогам обучения и является обязательной.

2. Критерии и шкала оценивания

<i>Критерии оценивания</i>	<i>Шкала оценивания по системе бальной оценки</i>			
	<i>неудовлетворительно</i>	<i>удовлетворительно</i>	<i>хорошо</i>	<i>отлично</i>
Полнота знаний	Уровень знаний ниже минимальных требований. Имели место грубые ошибки	Минимально допустимый уровень знаний. Допущено много негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки. Допущено несколько негрубых ошибок	Уровень знаний в объеме, соответствующем программе подготовки, без ошибок
Наличие умений	При решении стандартных задач не продемонстрированы основные умения. Имели место грубые ошибки	Продемонстрированы основные умения. Решены типовые задачи с негрубыми ошибками. Выполнены все задания, но не в полном объеме.	Продемонстрированы все основные умения. Решены все основные задачи с негрубыми ошибками. Выполнены все задания, в полном объеме, но некоторые с недочетами	Продемонстрированы все основные умения, решены все основные задачи с отдельными незначительными недочетами, выполнены все задания в полном объеме
Наличие навыков (владение опытом)	При решении стандартных задач не продемонстрированы базовые навыки. Имели место грубые ошибки	Имеется минимальный набор навыков для решения стандартных задач с некоторыми недочетами	Продемонстрированы базовые навыки при решении стандартных задач с некоторыми недочетами	Продемонстрированы навыки при решении нестандартных задач без ошибок и недочетов
Характеристика сформированности компетенции	Компетенция в полной мере не сформирована. Имеющихся знаний, умений, навыков недостаточно для решения профессиональ	Сформированность компетенции соответствует минимальным требованиям. Имеющихся знаний, умений, навыков в	Сформированность компетенции в целом соответствует требованиям, но есть недочеты. Имеющихся знаний, умений,	Сформированность компетенции полностью соответствует требованиям. Имеющихся знаний, умений, навыков и

Критерии оценивания	Шкала оценивания по системе бальной оценки			
	неудовлетворительно	удовлетворительно	хорошо	отлично
	ных задач. Требуется повторное обучение	целом достаточно для решения профессиональных задач, но требуется дополнительная практика по большинству практических задач	навыков и мотивации в целом достаточно для решения профессиональных задач, но требуется дополнительная практика по некоторым профессиональным задачам	мотивации в полной мере достаточно для решения сложных профессиональных задач
Уровень сформированности компетенций	Низкий	Ниже среднего	Средний	Высокий

3. Оценочные средства

3.1. Текущий контроль

3.1.1. Контролируемый раздел дисциплины «Язык программирования Python»

1. Как создать функцию в Python?
2. Как импортировать модуль в Python?
3. Как сортировать список в Python?
4. Каким методом можно удалить дубликаты из списка в Python?
5. Как форматировать строку в Python?
6. Как создать кортеж в Python?
7. Как выполнить клонирование или копирование списка в Python?
8. Каким методом можно добавить элемент в определенную позицию списка в Python?
9. Каким образом можно работать с данными в формате JSON в Python?
10. Каким методом можно выполнить замену подстроки в строке в Python?
11. Какие типы данных поддерживает Python?
12. Как объявить переменную в Python?
13. Какие операторы сравнения поддерживает Python?
14. Каким оператором можно проверить, содержится ли символ в строке?
15. Каким оператором можно выполнить целочисленное деление в Python?
16. Как выполнять итерации по элементам списка в Python?

17. Как определить, содержит ли строка только буквы или цифры в Python?
18. Каким оператором можно проверить, равен ли объект заданному значению или другому объекту?
19. Каким оператором можно проверить тип данных?
20. Как определить тип переменной в Python?

Тестовые вопросы к разделу «Язык программирования Python»

<i>Тестовые вопросы и варианты ответов</i>	<i>Компетенция, формируемая тестовым вопросом</i>
1. Какие типы данных поддерживает Python? 1) Строки, числа, списки, кортежи, словари, множества 2) Строки, числа, списки, кортежи, словари, деревья 3) Строки, числа, матрицы, словари, множества, функции 4) Строки, числа, массивы, кортежи, словари, множества	ОПК-2
2. Как объявить переменную в Python? 1) <code>x = 5</code> 2) <code>x := 5</code> 3) <code>declare x = 5</code> 4) <code>var x = 5</code>	ОПК-2
3. Какие операторы сравнения поддерживает Python? 1) <code>==, !=, >, <, >=, <=</code> 2) <code>==, <, >, <=, >=, <=</code> 3) <code>==, <, >, =>, =<</code> 4) <code>==, !=, >, <, <=, /></code>	ОПК-2
4. Каким оператором можно проверить, содержится ли символ в строке? 1) <code>-in-</code> 2) <code>contains</code> 3) <code>includes</code> 4) <code>in</code>	ОПК-5
5. Каким оператором можно выполнить целочисленное деление в Python? 1) <code>/</code> 2) <code>//</code> 3) <code>÷</code> 4) <code>:</code>	ОПК-5
6. Как выполнять итерации по элементам списка в	ОПК-5

<p>Python?</p> <ol style="list-style-type: none"> 1) for each 2) for i in list 3) foreach i in list 4) while i in list 	
<p>7. Как определить, содержит ли строка только буквы или цифры в Python?</p> <ol style="list-style-type: none"> 1) string.isalpha() 2) string.isdigit() 3) string.isalnum() 4) string.isnumeric() 	ОПК-2
<p>8. Каким оператором можно проверить, равен ли объект заданному значению или другому объекту?</p> <ol style="list-style-type: none"> 1) == 2) = 3) <> 4) ! 	ОПК-2
<p>9. Каким оператором можно проверить, не равно ли объект заданному значению или другому объекту?</p> <ol style="list-style-type: none"> 1) != 2) << 3) <> 4) != 	ПК-3
<p>10. Как определить тип переменной в Python?</p> <ol style="list-style-type: none"> 1) type(var) 2) typeof(var) 3) var.type() 4) var typeof() 	ОПК-5
<p>11. Как создать функцию в Python?</p> <ol style="list-style-type: none"> 1) function name(): 2) def name(): 3) func(name): 4) create function name(): 	ПК-3
<p>12. Как импортировать модуль в Python?</p> <ol style="list-style-type: none"> 1) import 2) use 3) from 4) include 	ПК-3

<p>13. Как сортировать список в Python?</p> <ol style="list-style-type: none">1) list.sort()2) sort(list)3) list.order()4) order(list)	ПК-3
<p>14. Каким методом можно удалить дубликаты из списка в Python?</p> <ol style="list-style-type: none">1) list.filter_duplicates()2) unique(list)3) remove_duplicates(list)4) list(set(list))	ОПК-5
<p>15. Как форматировать строку в Python?</p> <ol style="list-style-type: none">1) string.format()2) format.string()3) string.formatting()4) formatting.string()	ОПК-2
<p>16. Как создать кортеж в Python?</p> <ol style="list-style-type: none">1) [1, 2, 3]2) (1, 2, 3)3) {1, 2, 3}4) (1; 2; 3)	ОПК-5
<p>17. Как выполнить клонирование или копирование списка в Python?</p> <ol style="list-style-type: none">1) list.clone()2) copy(list)3) list.copy()4) clone(list)	ПК-3
<p>18. Каким методом можно добавить элемент в определенную позицию списка в Python?</p> <ol style="list-style-type: none">1) list.append()2) list.add()3) list.insert()4) list.push()	ПК-3
<p>19. Каким образом можно работать с данными в формате JSON в Python?</p> <ol style="list-style-type: none">1) import json; json.dumps()2) import json; json.decode()3) import json; json.load()4) import json; json.json()	ОПК-2

<p>20. Каким методом можно выполнить замену подстроки в строке в Python?</p> <ol style="list-style-type: none"> 1) <code>string.replace()</code> 2) <code>string.swap()</code> 3) <code>string.modify()</code> 4) <code>string.change()</code> 	ОПК-2
<p>21. Какие операторы являются логическими в Python?</p> <ol style="list-style-type: none"> 1) <code>=, ==, !=</code> 2) <code>and, or, not</code> 3) <code>+, -, *, /</code> 4) <code>+=, -=, *=, /=</code> 	ОПК-2
<p>22. Как перевернуть список в Python?</p> <ol style="list-style-type: none"> 1) <code>reverse()</code> 2) <code>flip()</code> 3) <code>turn()</code> 4) <code>swap()</code> 	ОПК-2
<p>23. Как создать множество в Python?</p> <ol style="list-style-type: none"> 1) <code>{1, 2, 3}</code> 2) <code>set(1, 2, 3)</code> 3) <code>(1, 2, 3)</code> 4) <code>[1, 2, 3]</code> 	ОПК-5
<p>24. Как объединить два словаря в Python?</p> <ol style="list-style-type: none"> 1) <code>dict1 + dict2</code> 2) <code>dict1.merge(dict2)</code> 3) <code>dict(dict1, dict2)</code> 4) <code>dict1.update(dict2)</code> 	ОПК-5
<p>25. Как определить количество элементов в списке в Python?</p> <ol style="list-style-type: none"> 1) <code>len(list)</code> 2) <code>count(list)</code> 3) <code>length(list)</code> 4) <code>items(list)</code> 	ПК-3
<p>26. Как удалить элемент из словаря в Python?</p> <ol style="list-style-type: none"> 1) <code>remove()</code> 2) <code>delete()</code> 3) <code>pop()</code> 4) <code>exclude()</code> 	ПК-3
<p>27. Как определить наличие значения в словаре в Python?</p>	ОПК-2

1) has() 2) in() 3) find() 4) exists()	
28. Как выполнить добавление элемента в конец списка в Python? 1) append() 2) insert() 3) push() 4) add()	ОПК-2
29. Как сделать первую букву строки заглавной в Python? 1) string.upper() 2) string.lower() 3) string.capitalize() 4) string.title()	ОПК-5
30. Как отсортировать список по убыванию в Python? 1) list.sort() 2) sorted(list) 3) list.sort(reverse=True) 4) sorted(list, reverse=True)	ОПК-5

Эталоны ответов

<i>Номер тестового задания</i>	<i>Номер эталона ответа</i>
1	1
2	1
3	1
4	4
5	2
6	2
7	3
8	1
9	4
10	1
11	2
12	1
13	1

14	1
15	1
16	2
17	3
18	3
19	1
20	1
21	1
22	2
23	2
24	1
25	1
26	3
27	4
28	1
29	3
30	3

1.1.1. Контролируемый раздел дисциплины «Язык SQL»

1. Применение баз данных в медицине. Идея базы данных. Назначение СУБД SQLite и программы SQLite Browser.
2. Структура базы данных. Таблицы и поля. Типы данных.
3. Атрибуты столбцов: ограничения, проверки и первичные ключи.
4. Синтаксис SQL-запросов на создание, переименование и удаление таблицы.
5. Синтаксис SQL-запросов на добавление одной и нескольких записей, на редактирование и удаление записей.
6. Синтаксис SQL-запросов на однотабличную выборку. Разновидности логических условий фильтрации и используемых в них операторов. Сортировка записей.
7. Синтаксис SQL-запросов на многотабличную выборку. Связывание таблиц. Комбинация связывания и фильтрации. Ограничение количества записей в результате выборки.
8. Синтаксис SQL-запросов на выборку с группировкой. Разновидности агрегатных функций и их использование для получения статистики по группам.
9. Синтаксис SQL-запросов для обеспечения связанности и целостности данных. Экспорт и импорт данных. Внешние ключи. Транзакции.
10. Синтаксис SQL-запросов встроенных средств языка SQL для обработки данных. Простые арифметические операции. Функции SQL для обработки данных.
11. Выполнение SQL-запросов из программы на языке Python.
12. Особенности работы с клиент-серверной СУБД PostgreSQL. Распределенные системы.

13. Триггеры в БД и их применение для мониторинга критических показателей жизнедеятельности организма.

3.1 Вопросы для экзамена

1. Применение баз данных в медицине. Идея базы данных и ее отличие от Excel-таблиц. Назначение СУБД SQLite и программы SQLite Browser.
2. Общая характеристика этапов построения и использования базы данных.
3. Структура базы данных. Таблицы и поля. Типы данных. Особенности и форматы хранения номеров телефонов, номеров документов и дат в БД.
4. Атрибуты столбцов: ограничения, проверки и первичные ключи.
5. Синтаксис SQL-запросов на создание, переименование и удаление таблицы.
6. Синтаксис SQL-запросов на добавление одной и нескольких записей, на редактирование и удаление записей. Условия их безопасного выполнения.
7. Синтаксис SQL-запросов на однотабличную выборку. Разновидности логических условий фильтрации и используемых в них операторов. Сортировка записей. Ограничение количества записей в результате выборки.
8. Синтаксис SQL-запросов на многотабличную выборку. Связывание таблиц. Комбинация связывания и фильтрации. Сохранение запроса на выборку в виде представления, его назначение и использование.
9. Синтаксис SQL-запросов на выборку с группировкой. Разновидности агрегатных функций и их использование для получения статистики по группам.
10. Проведение арифметических вычислений в SQL, округление. Особенности операций умножения и деления с целочисленными и дробными данными. Генерация случайных чисел.
11. Функции языка SQL для работы с текстовыми данными. Оператор склейки.
12. Функции языка SQL для работы с датой и временем. Получение текущей даты и времени с часов компьютера. Получение даты и времени со смещением в прошлое или будущее.
13. Экспорт и импорт данных. Добавление нового столбца в середину структуры таблицы. Резервное копирование данных.
14. Внешние ключи и их использование для связывания таблиц. Отличия внешнего ключа от первичного и их взаимосвязь.
15. Транзакции и их назначение. Простая транзакция, транзакция с точками сохранения. Применение и откат изменений БД в рамках транзакции.
16. Синтаксические и логические ошибки при составлении SQL-запросов, методы их поиска и устранения.
17. Библиотека sqlite3 языка Python. Соединение с БД. Подача запросов в БД и получение результата выборки при помощи курсора. Применение изменений.
18. Разграничение доступа в СУБД PostgreSQL. Идентификация, аутентификация и авторизация.
19. Дополнительные типы данных в PostgreSQL. Ширина поля и ее значение при хранении больших объемов данных. Особенности работы некоторых функций в PostgreSQL по сравнению с SQLite.
20. Триггеры в БД и их применение для мониторинга критических показателей жизнедеятельности организма.

Тестовые вопросы к разделу «Язык SQL»

<i>Тестовые вопросы и варианты ответов</i>	<i>Компетенция, формируемая тестовым вопросом</i>
<p>1. ЧТО ТАКОЕ РЕЛЯЦИОННЫЕ БАЗЫ ДАННЫХ:</p> <p>1) База данных, в которой информация хранится в виде двумерных таблиц, связанных между собой</p> <p>2) База данных, в которой одна ни с чем не связанная таблица</p> <p>3) Любая база данных - реляционная</p> <p>4) Совокупность данных, не связанных между собой</p>	ОПК-2
<p>2. КАК ВЫГЛЯДИТ ЗАПРОС, ДЛЯ ВЫВОДА ВСЕХ ЗНАЧЕНИЙ ИЗ ТАБЛИЦЫ ДОНОРЫ:</p> <p>1) select ALL from Доноры;</p> <p>2) select % from Доноры;</p> <p>3) select * from Доноры;</p> <p>4) select *.Orders from Доноры;</p>	ОПК-2
<p>3. ЕСТЬ ЛИ ОШИБКА В ЗАПРОСЕ?</p> <p>SELECT имя_донора, дата_донации FROM доноры WHERE имя_донора = Иван;</p> <p>1) Запрос составлен правильно</p> <p>2) Иван необходимо записать в кавычках 'Иван'</p> <p>3) Нужно убрать лишние поля из запроса</p> <p>4) Строчку с where поменять местами с from</p>	ОПК-2
<p>4. ЧТО ПОКАЖЕТ СЛЕДУЮЩИЙ ЗАПРОС:</p> <p>SELECT * FROM Донации WHERE дата_донации BETWEEN '2017-01-01' AND '2017-12-31'?</p> <p>1) Все данные по донациям, совершенным за 2017 год, за исключением 01 января 2017 года</p> <p>2) Все данные по донациям, совершенным за 2017 год, за исключением 31 декабря 2017 года</p> <p>3) Все данные по донациям, совершенным за 2017 год</p> <p>4) Ничего, запрос составлен неверно</p>	ОПК-5
<p>5. ЧТО НЕ ТАК С ЭТИМ ЗАПРОСОМ?</p> <p>SELECT номер_донора, дата_анализа FROM анализы WHERE дата_анализа = NULL?</p> <p>1) NULL нужно взять в кавычки</p> <p>2) Сравнение с NULL можно проводить только с оператором IS</p> <p>3) Сравнение с NULL можно проводить только с оператором ON</p>	ОПК-5

4) Сравнение с NULL можно проводить только с оператором IN	
<p>6. ПОРЯДОК ВЫПОЛНЕНИЯ ОПЕРАТОРОВ AND И OR СЛЕДУЮЩИЙ:</p> <ol style="list-style-type: none"> 1) Сначала выполняется AND, а затем OR 2) Сначала выполняется OR, а затем AND 3) Порядок выполнения операторов AND и OR зависит от того, какой операторов стоит первым 4) Операторы AND и OR выполняются одновременно 	ОПК-5
<p>7. ЧТО ПОКАЖЕТ СЛЕДУЮЩИЙ ЗАПРОС: SELECT DISTINCT НОМЕР_ДОНОРА ORDER BY НОМЕР_ДОНОРА FROM ДОНОРЫ;</p> <ol style="list-style-type: none"> 1) Уникальные ID доноров, отсортированные по возрастанию 2) Уникальные ID доноров, отсортированные по убыванию 3) Ничего, запрос составлен неверно, ORDER BY всегда ставится в конце запроса 4) Неотсортированные никак уникальные ID доноров 	ОПК-2
<p>8. ЧТО ДЕЛАЕТ СПЕЦСИМВОЛ ' _ ' В ПАРЕ С ОПЕРАТОРОМ LIKE: SELECT * FROM ДОНОРЫ WHERE ИМЯ_ДОНОРА LIKE 'ИВА_';</p> <ol style="list-style-type: none"> 1) найдет все имена, которые начинаются на ИВА и состоят из 4 символов 2) найдет все имена, которые начинаются на ИВА, вне зависимости от того, из какого количества символов они состоят 3) найдет данные, где имя равно ИВА 4) запрос составлен неверно, в паре с оператором like не используются спецсимволы 	ОПК-2
<p>9. ВЫБЕРИТЕ ПРАВИЛЬНЫЙ ПРИМЕР ИСПОЛЬЗОВАНИЯ ФУНКЦИИ ОКРУГЛЕНИЯ ROUND;</p> <ol style="list-style-type: none"> 1) select номер_донора, эритроциты * discount AS total эритроциты from Анализы ROUND (2); 2) select id, ROUND (эритроциты * discount, 2) AS total эритроциты from Анализы; 3) select номер_донора, эритроциты * discount from Анализы ROUND (2); 4) нет правильного примера 	ПК-3
10. ЧТО ПОКАЖЕТ СЛЕДУЮЩИЙ ЗАПРОС: SELECT	ОПК-5

<p>ID FROM ДОНАЦИИ WHERE ДАТА_ДОНАЦИИ (DATE) > 2018;</p> <ol style="list-style-type: none"> 1) номера донаций, сделанных до 2018 года 2) номера донаций, сделанных в 2018 году 3) номера донаций, сделанных после 2018 года 4) запрос написан не правильно 	
<p>11. ДЛЯ ЧЕГО ИСПОЛЬЗУЕТСЯ LIMIT: SELECT * FROM ДОНАЦИИ LIMIT 10;</p> <ol style="list-style-type: none"> 1) необходим, чтобы показать все донации, содержащие цифру 10 2) необходим, чтобы показать первых 10 записей в запросе 3) необходим, чтобы показать рандомные 10 записей в запрос 4) не существует такого оператора 	ПК-3
<p>12. ЧТО ТАКОЕ АГРЕГИРУЮЩИЕ ФУНКЦИИ:</p> <ol style="list-style-type: none"> 1) функции, которые фильтруют значения 2) функции, которые сортируют значения 3) функции, которые работают с набором данных, превращая их в одно итоговое значение 4) функции, которые объединяют данные 	ПК-3
<p>13. ВЫБЕРИТЕ ПРИМЕР ПРАВИЛЬНО СОСТАВЛЕННОГО ЗАПРОСА С ИСПОЛЬЗОВАНИЕМ АГРЕГИРУЮЩЕЙ ФУНКЦИИ SUM:</p> <ol style="list-style-type: none"> 1) select sum(донации_за_год) from донации; 2) select sum(донации_за_год), имя_донора from донации; 3) select * from донации where донации_за_год=sum(); 4) нет правильного ответа 	ПК-3
<p>14. ВОЗМОЖНО ЛИ ИСПОЛЬЗОВАНИЕ ОДНОВРЕМЕННО ДВУХ АГРЕГИРУЮЩИХ ФУНКЦИЙ: SELECT MIN(ДОНАЦИИ_ЗА_ГОД), MAX(ДОНАЦИИ_ЗА_ГОД) FROM ДОНАЦИИ;</p> <ol style="list-style-type: none"> 1) да, в результате мы получим минимальную и максимальную донации 2) да, в результате мы получим донации, отсортированные от минимальной к максимальной 3) нет, две функции использовать одновременно нельзя 4) нет правильного ответа 	ОПК-5
<p>15 ЧТО ТАКОЕ JOIN:</p>	ОПК-2

<ol style="list-style-type: none"> 1) операция объединения 2) операция группировки 3) операция суммирования 4) операция создания 	
<p>16. КАКОГО ИЗ ПЕРЕЧИСЛЕННЫХ НИЖЕ ВИДОВ JOIN НА САМОМ ДЕЛЕ НЕ СУЩЕСТВУЕТ:</p> <ol style="list-style-type: none"> 1) LEFT JOIN - который выведет все записи первой таблицы, а для ненайденных пар из правой таблицы проставит значение NULL 2) RIGHT JOIN - который выведет все записи второй таблицы, а на место недостающей информации из первой таблицы проставит NULL 3) INNER JOIN - который показывает только те записи, для которых нашлись пары 4) TRUE JOIN - который выведет все верные значения 	ОПК-5
<p>17. КАКОГО СТРОКОВОГО ТИПА ДАННЫХ НЕТ В SQL:</p> <ol style="list-style-type: none"> 1) VARCHAR 2) STRING 3) CHAR 4) TEXT 	ПК-3
<p>18. ЧЕМ ОТЛИЧАЕТСЯ CHAR И VARCHAR?</p> <ol style="list-style-type: none"> 1) Это одно и то же 2) VARCHAR не существует 3) CHAR - это тип данных, а VARCHAR - подтип 4) CHAR дополняет строку пробелами до максимальной длины, а VARCHAR тратит лишнюю память на хранение значения длины строки 	ПК-3
<p>19. КАК ПОЛУЧИТЬ ЗНАЧЕНИЕ ТЕКУЩЕГО ГОДА В SQL?</p> <ol style="list-style-type: none"> 1) select now(); 2) select year(); 3) select year(now()); 4) select year from Date; 	ОПК-2
<p>20. КАКИЕ ПОЛЯ ИЗ ТАБЛИЦЫ ОБЯЗАТЕЛЬНО ПЕРЕЧИСЛЯТЬ В INSERT ДЛЯ ВСТАВКИ ДАННЫХ?</p> <ol style="list-style-type: none"> 1) Только те, у которых нет DEFAULT значения 2) Те, у которых нет DEFAULT значения и которые не имеют атрибут auto_increment 3) Все поля имеют негласное DEFAULT значения, 	ОПК-2

<p>обязательных полей в SQL нет</p> <p>4) Любые поля</p>	
<p>21. КАК СДЕЛАТЬ НЕСКОЛЬКО ЗАПИСЕЙ В ТАБЛИЦУ ЗА ОДИН ЗАПРОС?</p> <p>1) Перечислить через запятую все наборы значений после VALUES</p> <p>2) Использовать подзапрос</p> <p>3) Использовать MULTI INSERT INTO вместо INSERT INTO</p> <p>4) Все вышеперечисленное</p>	ОПК-2
<p>22. ЗАЧЕМ СУЩЕСТВУЕТ КОМАНДА UPDATE, ЕСЛИ МОЖНО СНАЧАЛА УДАЛИТЬ ЗАПИСЬ, А ПОТОМ ДОБАВИТЬ НОВУЮ, ИСПРАВЛЕННУЮ.</p> <p>1) Так меньше нагрузки на базу, ведь команда одна, а не две</p> <p>2) Потому что в записи могут быть автоматически предоставляемые поля, такие как auto_increment или timestamp, которые сойдутся при внесении записи заново</p> <p>3) Как раз удалять записи в SQL нельзя, вместо этого используется UPDATE с NULL-значениями для всех полей</p> <p>4) Нет правильного ответа</p>	ОПК-2
<p>23. В КАКИХ КОМАНДАХ МОЖНО ИСПОЛЬЗОВАТЬ LIMIT?</p> <p>1) Только Select</p> <p>2) Select и Insert</p> <p>3) Select, Update, Delete</p> <p>4) Select, Insert, Delete, Update</p>	ОПК-5
<p>24. КАК МОЖНО ЗАРАНЕЕ УЗНАТЬ, КАКИЕ ЗАПИСИ БУДУТ УДАЛЕНЫ ПРИ ВЫПОЛНЕНИИ DELETE?</p> <p>1) Заменить DELETE на SELECT *, ведь в остальном синтаксис DELETE похож на синтаксис простого SELECT</p> <p>2) Сделать DELETE с LIMIT 1, одну запись</p> <p>3) SQL создан для хранения данных, их нельзя удалять</p> <p>4) Сделать DELETE с DISTINCT</p>	ОПК-5
<p>25. КАКОЙ КОМАНДОЙ МОЖНО СОЗДАТЬ НОВУЮ</p>	ПК-3

<p>ТАБЛИЦУ?</p> <ol style="list-style-type: none"> 1) CREATE TABLE 2) MAKE TABLE 3) SET TABLE 4) Создавать таблицы можно только через интерфейс СУБД, специальной SQL команды для этого нет 	
<p>26. МОЖНО ЛИ ПОМЕНЯТЬ ТИП ДАННЫХ ПОЛЯ В УЖЕ СУЩЕСТВУЮЩЕЙ ТАБЛИЦЕ?</p> <ol style="list-style-type: none"> 1) Да, при помощи команды ALTER 2) Да, достаточно сделать INSERT с новым типом данных 3) Нет, только пересоздать таблицу 4) Тип бывает только у таблицы, а не у поля таблицы 	ПК-3
<p>27. КАКАЯ ФУНКЦИЯ SQL ПОМОЖЕТ УЗНАТЬ, СКОЛЬКО ВРАЧЕЙ-ЭПИДЕМИОЛОГОВ ИМЕЕТСЯ НА ВЕСЬ ГОРОД?</p> <ol style="list-style-type: none"> 1) ABS() 2) COUNT() 3) ROUND() 4) AVG() 	ОПК-2
<p>28. В ТАБЛИЦЕ "ВЫЗОВЫ_НА_ДОМ" ВРАЧЕЙ-ТЕРАПЕВТОВ УЧАСТКОВЫХ ЕСТЬ НЕСКОЛЬКО ПОЛЕЙ С ИДЕНТИФИКАТОРАМИ. КАКОЕ ИЗ НИХ НУЖНО ВЫБРАТЬ ПЕРВИЧНЫМ КЛЮЧОМ?</p> <ol style="list-style-type: none"> 1) "id_участка" 2) "id_пациента" 3) "id_вызова" 4) "id_врача" 	ОПК-2
<p>29. ПОСЛЕ КАЖДОГО ОБНОВЛЕНИЯ ЗАПИСИ ОБ АД ПАЦИЕНТА ПРОИЗВОДИТСЯ СОХРАНЕНИЕ СОСТОЯНИЯ ПОД ИМЕНЕМ СОСТОЯНИЕ_N, ГДЕ N - ПОРЯДКОВЫЙ НОМЕР ИЗМЕРЕНИЯ. НА РАБОТУ ПРИШЛА НЕОПЫТНАЯ МЕДСЕСТРА И НЕПРАВИЛЬНО НАДЕЛА МАНЖЕТУ НА РУКУ ПАЦИЕНТА. КАК ОТМЕНИТЬ ПОСЛЕДНИЕ 2 ИЗМЕНЕНИЯ, ВОССТАНОВИВ КОРРЕКТНУЮ ИНФОРМАЦИЮ?</p> <ol style="list-style-type: none"> 1) ROLLBACK TO Состояние_N-2 	ОПК-5

2) SAVEPOINT Состояние_N-2 3) RELEASE Состояние_N-2 4) COMMIT Состояние_N-2	
30. КАКОЙ ЗАПРОС СЛЕДУЕТ ИСПОЛЬЗОВАТЬ ДЛЯ ПЕРЕИМЕНОВАНИЯ ТАБЛИЦЫ "ТАЛОНЫНАКОНСУЛЬТАЦИЮ" В "ТАЛОНЫНАПРИЕМ"? 1) Rename table "ТалоныНаКонсультацию" to "ТалоныНаПрием"? 2) update table "ТалоныНаКонсультацию" to "ТалоныНаПрием"? 3) update table "ТалоныНаКонсультацию" in "ТалоныНаПрием"? 4) set table "ТалоныНаКонсультацию" values "ТалоныНаПрием"?	ОПК-5

Эталоны ответов

<i>Номер тестового задания</i>	<i>Номер эталона ответа</i>
1	1
2	3
3	2
4	3
5	2
6	1
7	3
8	1
9	2
10	3
11	2
12	3
13	1
14	1
15	1
16	4
17	2

18	4
19	3
20	2
21	1
22	2
23	3
24	1
25	1
26	1
27	2
28	1
29	1
30	1

1.1.2. Контролируемый раздел дисциплины «Язык PHP, html, JavaScript»

1. Какие основные теги HTML вы знаете и для чего они используются?
2. Стандарты языка HTML. Необходимые программные средства. Элементы HTML. Структура документа. Форматирование текста. Списки.
3. Язык разметки гипертекста HTML. Гиперссылки и рисунки. Таблицы. Фреймы.
4. Как вставить изображение на веб-страницу с помощью HTML?
5. Что такое CSS и как он связан с HTML?
6. Что такое селектор в CSS и какие виды селекторов вы знаете
7. Что такое псевдоэлемент в CSS и для чего он используется?
8. Каким образом можно задать размеры элемента в CSS?
9. Дополнительные возможности при создании Web страниц. Применение сценариев (JavaScript, VBScript) и CGI.
10. Характеристика CMS. Классификация и примеры.
11. WEB 2.0 Общая характеристика и примеры.
12. Формы. Организация форм на Web-странице.
13. Возможности языка PHP
14. Технология клиент-сервер в PHP
15. Какой оператор используется для выполнения условных операций в PHP? Приведите пример.
16. Какой цикл можно использовать для перебора элементов массива в PHP? Приведите пример.
17. Назначение и возможности JavaScript.

18. Как объявить переменную в JavaScript и какие типы данных поддерживает этот язык программирования?
19. Какие условные операторы существуют в JavaScript и как они работают?
20. Что такое объекты в JavaScript и какие свойства и методы они могут иметь?

Тестовые вопросы к разделу «Язык PHP, html, JavaScript»

<i>Тестовые вопросы и варианты ответов</i>	<i>Компетенция, формируемая тестовым вопросом</i>
1. Какое расширение у файлов HTML? 1) .htmlx 2) .htm 3) .html 4) .ht	ОПК-2
2. Каким тегом обозначается заголовок первого уровня? 1) <h1> 2) <h2> 3) <h3> 4) <h4>	ОПК-2
3. Какой тег используется для создания ссылки? 1) <a> 2) <link> 3) <href> 4) <url>	ОПК-2
4. Каким атрибутом задается путь к файлу изображения? 1) path 2) href 3) src 4) url	ОПК-5
5. Какой тег используется для создания списка? 1) 2) 3) 4) <dl>	ОПК-5
6. Как изменить цвет текста в CSS? 1) text-color	ОПК-5

<p>2) font-color 3) color 4) Ни один из перечисленных</p>	
<p>7. Как задать границу для блока в CSS? 1) border-style 2) outline-style 3) box-style 4) Ни один из перечисленных</p>	ОПК-2
<p>8. Как сделать шрифт жирным в CSS? 1) font-style: bold 2) font-weight: bold 3) font-size: bold 4) Ни один из перечисленных</p>	ОПК-2
<p>9. ЧТО НАЗЫВАЕТСЯ ПЕРЕМЕННОЙ? 1) специальный участок памяти, в котором программа может сохранять данные 2) специальный тэг, значение которого можно изменить 3) параметр объекта, значение которого можно изменить 4) Ни один из перечисленных</p>	ПК-3
<p>10. КАКОЕ КЛЮЧЕВОЕ СЛОВО ПОЗВОЛЯЕТ ОПИСАТЬ ПЕРЕМЕННУЮ В HTML? 1) var 2) set 3) create 4) Ни один из перечисленных</p>	ОПК-5
<p>11. КАКАЯ КОМАНДА В HTML ВЫВОДИТ ОКНО С СООБЩЕНИЕМ И ПОЛЕМ ВВОДА, И ПОЗВОЛЯЕТ ВВЕСТИ ДАННЫЕ В ПЕРЕМЕННУЮ? 1) readln 2) prompt 3) insert 4) Ни один из перечисленных</p>	ПК-3
<p>12.Какой тег используется для создания таблицы в CSS? 1) <table> 2) <tr></p>	ПК-3

<p>3) <td></p> <p>4) Ни один из перечисленных</p>	
<p>13. Как сделать текст курсивом в CSS?</p> <p>1) font-style: italic</p> <p>2) font-weight: italic</p> <p>3) font-size: italic</p> <p>4) Ни один из перечисленных</p>	ПК-3
<p>14. Как изменить фоновый цвет элемента в CSS?</p> <p>1) bg-color</p> <p>2) background-color</p> <p>3) color-background</p> <p>4) Ни один из перечисленных</p>	ОПК-5
<p>15. Что означает аббревиатура "PHP"?</p> <p>1) Personal Home Page</p> <p>2) Pre Hypertext Processor</p> <p>3) Hypertext Preprocessor</p> <p>4) Private Home Page</p>	ОПК-2
<p>16. Какой оператор используется для объявления переменной в PHP?</p> <p>1) %</p> <p>2) &</p> <p>3) \$</p> <p>4) @</p>	ОПК-5
<p>17. НАЗОВИТЕ ЭЛЕМЕНТ В HTML, ПРЕДНАЗНАЧЕННЫЙ ДЛЯ РАЗМЕТКИ ВРЕМЕНИ И ДАТЫ?</p> <p>1) <date></p> <p>2) <div></p> <p>3) <time></p> <p>4) <datetime></p>	ПК-3
<p>18. ЭЛЕМЕНТЫ ФОРМЫ КАКОГО ТИПА ПРЕДОСТАВЛЯЮТ ПОЛЬЗОВАТЕЛЮ ПАЛИТРУ ДЛЯ ВЫБОРА ЦВЕТА В HTML?</p> <p>1) range</p> <p>2) text</p> <p>3) color</p> <p>4) background</p>	ПК-3

5) number	
<p>19. Какой тип данных в PHP предназначен для хранения текстовых значений?</p> <p>1) integer 2) string 3) float 4) boolean</p>	ОПК-2
<p>20. Какой оператор в PHP используется для сравнения значений и типов данных?</p> <p>1) == 2) === 3) = 4) <=></p>	ОПК-2
<p>21. Какой цикл в PHP позволяет выполнять определенный блок кода до тех пор, пока условие истинно?</p> <p>1) for 2) while 3) foreach 4) do-while</p>	ОПК-2
<p>22. Какой оператор в PHP используется для объединения двух строк?</p> <p>1) + 2) - 3) * 4) .</p>	ОПК-2
<p>23. Какой из следующих вариантов правильно определяет функцию в PHP?</p> <p>1) function name() {} 2) function(name) {} 3) function name {} 4) function [name] {}</p>	ОПК-5
<p>24. Какой массив в PHP используется для хранения информации, полученной через форму на веб-странице?</p> <p>1) \$_GET 2) \$_POST 3) \$_REQUEST</p>	ОПК-5

4) \$_SESSION	
25. Что означает аббревиатура DOM? 1) Document Original Model 2) Document Object Model 3) Document Offline Mode 4) Document Object Mapping	ПК-3
26. Каким образом можно создать новый массив в JavaScript? 1) var arr = (1, 2, 3); 2) var arr = [1, 2, 3]; 3) var arr = {1, 2, 3}; 4) var arr = "1, 2, 3";	ПК-3
27. Что вернет оператор typeof для строки "123"? 1) "number" 2) "string" 3) "object" 4) "undefined"	ОПК-2
28. Какая функция используется для добавления элемента в конец массива в JavaScript? 1) arr.push() 2) arr.shift() 3) arr.pop() 4) arr.unshift()	ОПК-2
29. Какой метод можно использовать для преобразования строки в число? 1) parseInt() 2) toInteger() 3) toNumber() 4) toString()	ОПК-5
30. Какой оператор используется для проверки на строгое равенство? 1) == 2) != 3) === 4) !==	ОПК-5

Эталоны ответов

<i>Номер тестового задания</i>	<i>Номер эталона ответа</i>
--------------------------------	-----------------------------

1	2
2	1
3	1
4	3
5	3
6	3
7	1
8	2
9	1
10	1
11	2
12	4
13	1
14	2
15	3
16	3
17	4
18	3
19	2
20	2
21	2
22	4
23	1
24	2
25	2
26	2
27	2
28	1
29	1
30	3

3.1 Вопросы для экзамена

21. Применение баз данных в медицине. Идея базы данных и ее отличие от Excel-таблиц. Назначение СУБД SQLite и программы SQLite Browser.
22. Общая характеристика этапов построения и использования базы данных.
23. Структура базы данных. Таблицы и поля. Типы данных. Особенности и форматы хранения номеров телефонов, номеров документов и дат в БД.
24. Атрибуты столбцов: ограничения, проверки и первичные ключи.
25. Синтаксис SQL-запросов на создание, переименование и удаление таблицы.
26. Синтаксис SQL-запросов на добавление одной и нескольких записей, на редактирование и удаление записей. Условия их безопасного выполнения.
27. Синтаксис SQL-запросов на однотабличную выборку. Разновидности логических условий фильтрации и используемых в них операторов. Сортировка записей. Ограничение количества записей в результате выборки.
28. Синтаксис SQL-запросов на многотабличную выборку. Связывание таблиц. Комбинация связывания и фильтрации. Сохранение запроса на выборку в виде представления, его назначение и использование.
29. Синтаксис SQL-запросов на выборку с группировкой. Разновидности агрегатных функций и их использование для получения статистики по группам.
30. Проведение арифметических вычислений в SQL, округление. Особенности операций умножения и деления с целочисленными и дробными данными. Генерация случайных чисел.
31. Функции языка SQL для работы с текстовыми данными. Оператор склейки.
32. Функции языка SQL для работы с датой и временем. Получение текущей даты и времени с часов компьютера. Получение даты и времени со смещением в прошлое или будущее.
33. Экспорт и импорт данных. Добавление нового столбца в середину структуры таблицы. Резервное копирование данных.
34. Внешние ключи и их использование для связывания таблиц. Отличия внешнего ключа от первичного и их взаимосвязь.
35. Транзакции и их назначение. Простая транзакция, транзакция с точками сохранения. Применение и откат изменений БД в рамках транзакции.
36. Синтаксические и логические ошибки при составлении SQL-запросов, методы их поиска и устранения.
37. Библиотека sqlite3 языка Python. Соединение с БД. Подача запросов в БД и получение результата выборки при помощи курсора. Применение изменений.
38. Разграничение доступа в СУБД PostgreSQL. Идентификация, аутентификация и авторизация.
39. Дополнительные типы данных в PostgreSQL. Ширина поля и ее значение при хранении больших объемов данных. Особенности работы некоторых функций в PostgreSQL по сравнению с SQLite.
40. Триггеры в БД и их применение для мониторинга критических показателей жизнедеятельности организма.