

Вопросы к экзамену по веб-разработке

Большой набор закрытых вопросов по теории: сети, HTTP, HTML, CSS, Django models, ORM, URLs, views, templates и forms. Правильные ответы вынесены в ключ в конце файла.

Всего вопросов: 144

Формат: A-D

Ответы: ключ в конце

Блок 1. Сети, адресация и протоколы

1. Что в вебе указывает конкретный узел сети?

- A. HTTP-метод
- B. IP-адрес
- C. HTML-тег
- D. CSS-селектор

2. Сколько бит в IPv4-адресе?

- A. 16
- B. 32
- C. 64
- D. 128

3. Сколько бит в IPv6-адресе?

- A. 32
- B. 64
- C. 96
- D. 128

4. Какой адрес обычно означает localhost?

- A. 0.0.0.0
- B. 8.8.8.8
- C. 127.0.0.1
- D. 192.168.1.255

5. Какой диапазон относится к приватным IPv4-адресам?

- A. 8.8.8.0/24
- B. 10.0.0.0/8
- C. 1.1.1.0/24
- D. 224.0.0.0/4

6. Что показывает маска подсети?

- A. Порт приложения
- B. Где часть сети, а где часть хоста
- C. Версию HTTP
- D. Тип HTML-документа

7. Что означает CIDR-запись /24?

- A. 24 хоста
- B. 24 порта
- C. 24 сетевых бита
- D. 24 домена

8. Что делает NAT?

- A. Шифрует HTTP
- B. Переводит адреса на границе сети
- C. Создаёт HTML-форму
- D. Считает CSS-специфичность

9. Что делает PAT?

- A. Различает подключения по портам при NAT
- B. Проверяет пароль
- C. Создаёт DNS-запись
- D. Удаляет cookies

10. Что выбирает порт?

- A. Устройство в интернете
- B. Приложение внутри IP-адреса
- C. Доменную зону
- D. CSS-класс

11. Какой порт обычно использует HTTP?

- A. 21
- B. 22
- C. 80
- D. 443

12. Какой порт обычно использует HTTPS?

- A. 25
- B. 53
- C. 110
- D. 443

Блок 2. TCP, DNS, URI и OAuth

13. Какой первый шаг TCP-рукопожатия?

- A. ACK
- B. SYN
- C. FIN
- D. RST

14. Что делает TCP?

- A. Передаёт данные без подтверждений
- B. Контролирует доставку и порядок данных
- C. Описывает HTML-разметку
- D. Назначает CSS-стили

15. Что характерно для UDP?

- A. Обязательное соединение
- B. Гарантированный порядок
- C. Нет встроенной гарантии доставки
- D. Работает только с HTML

16. Что делает DNS?

- A. Превращает доменное имя в IP-адрес
- B. Шифрует страницу
- C. Валидирует форму
- D. Сортирует QuerySet

17. Какая DNS-запись указывает IPv4-адрес?

- A. AAAA
- B. MX
- C. A
- D. TXT

18. Какая DNS-запись указывает IPv6-адрес?

- A. AAAA
- B. CNAME
- C. MX
- D. NS

19. Что такое URL?

- A. Только имя без расположения
- B. Адрес и способ получения ресурса
- C. CSS-правило
- D. Django-модель

20. Что из перечисленного является фрагментом URL?

- A. https
- B. example.com
- C. ?page=2
- D. #comments

21. Верно ли: каждый URL является URI?

- A. Да
- B. Нет
- C. Только для HTTPS
- D. Только в Django

22. Что такое идентификация?

- A. Проверка прав
- B. Пользователь сообщает, кто он
- C. Шифрование запроса
- D. Создание токена CSS

23. Что такое авторизация?

- A. Проверка, что пользователю разрешено делать
- B. Ввод логина
- C. Передача HTML
- D. Расчёт маски

24. Что выдаёт OAuth 2.0 стороннему приложению вместо пароля?

- A. CSS-класс
- B. Access token
- C. HTML-id
- D. SQL JOIN

Блок 3. HTTP, HTTPS и HTTP/3

25. Из чего состоит HTTP-обмен?

- A. Запрос и ответ
- B. Только ответ
- C. Только HTML
- D. Только IP-пакет

26. Какой HTTP-метод обычно используется для получения ресурса?

- A. POST
- B. DELETE
- C. GET
- D. PATCH

27. Какой HTTP-метод обычно отправляет данные формы в теле запроса?

- A. GET
- B. POST
- C. HEAD
- D. OPTIONS

28. Какой класс кодов означает успешную обработку?

- A. 1xx
- B. 2xx
- C. 4xx
- D. 5xx

29. Какой код обычно означает, что ресурс не найден?

- A. 200
- B. 301
- C. 404
- D. 500

30. Какой код обычно означает внутреннюю ошибку сервера?

- A. 100
- B. 204
- C. 403
- D. 500

31. Что добавляет HTTPS к HTTP?

- A. CSS Grid
- B. TLS-защиту
- C. Django ORM
- D. HTML-таблицу

32. Что проверяет TLS-сертификат сервера?

- A. CSS-специфичность
- B. Что сервер действительно тот, за кого себя выдаёт
- C. Количество HTML-тегов
- D. Количество строк в таблице

33. Поверх чего работает HTTP/3?

- A. QUIC поверх UDP
- B. Только TCP
- C. HTML
- D. DNS TXT

34. Какой заголовок содержит сведения о браузере клиента?

- A. Host
- B. User-Agent
- C. Content-Type
- D. Set-Cookie

35. Какой заголовок обычно указывает тип тела запроса или ответа?

- A. Content-Type
- B. Accept-Language
- C. Location
- D. Path

36. Что обычно содержит заголовок Authorization: Bearer ...?

- A. Путь к CSS
- B. Токен доступа
- C. HTML-комментарий
- D. SQL-запрос

Блок 4. HTML: структура и семантика

37. Какой тег является корневым элементом HTML-документа?

- A. head
- B. body
- C. html
- D. main

38. Где находится видимый контент страницы?

- A. head
- B. body
- C. meta
- D. title

39. Какой тег задаёт название вкладки браузера?

- A. h1
- B. title
- C. header
- D. caption

40. Для чего нужен meta viewport?

- A. Для корректной ширины страницы на мобильных устройствах
- B. Для создания таблицы
- C. Для отправки формы
- D. Для CSS-анимации

41. Как пишется HTML-комментарий?

- A. // comment
- B. /* comment */
- C. <!-- comment →
- D. {# comment #}

42. Какой тег обычно содержит главную навигацию?

- A. nav
- B. aside
- C. footer
- D. code

43. Какой тег обозначает главный контент страницы?

- A. main
- B. span
- C. br
- D. meta

44. Какой тег подходит для самостоятельной статьи или поста?

- A. article
- B. div
- C. tbody
- D. input

45. Какой тег является нейтральным блочным контейнером?

- A. span
- B. div
- C. em
- D. img

46. Какой тег является нейтральным строчным контейнером?

- A. span
- B. section
- C. table
- D. form

47. Какой атрибут изображения обязателен для доступности?

- A. href
- B. alt
- C. method
- D. for

48. Какой атрибут у ссылки задаёт адрес перехода?

- A. src
- B. href
- C. alt
- D. name

49. Какой тег создаёт маркированный список?

- A. ol
- B. ul
- C. li
- D. dl

50. Какой тег создаёт нумерованный список?

- A. ol
- B. ul
- C. li
- D. span

51. Какой тег является элементом списка?

- A. li
- B. tr
- C. td
- D. th

52. Какой тег создаёт таблицу?

- A. table
- B. tbody
- C. caption
- D. fieldset

53. Какой тег создаёт строку таблицы?

- A. td
- B. tr
- C. th
- D. thead

54. Какой тег создаёт заголовочную ячейку таблицы?

- A. td
- B. tr
- C. th
- D. tfoot

55. Какой атрибут объединяет ячейки по горизонтали?

- A. rowspan
- B. colspan
- C. headers
- D. scope

56. Какой тег является контейнером HTML-формы?

- A. input
- B. button
- C. form
- D. label

57. Какой атрибут формы задаёт URL отправки?

- A. method
- B. action
- C. name
- D. target

58. Какой атрибут поля отправляется серверу как имя параметра?

- A. id
- B. name
- C. class
- D. for

59. Как связать label с input?

- A. Через href и src
- B. Через for у label и id у input
- C. Через class
- D. Через colspan

60. Что делает required?

- A. Скрывает поле
- B. Делает поле обязательным
- C. Сбрасывает форму
- D. Отправляет GET-запрос

Блок 6. HTML Forms: элементы и атрибуты

61. Какой input выбирает один вариант из группы?

- A. checkbox
- B. radio
- C. text
- D. hidden

62. Что нужно одинаковое у radio-кнопок одной группы?

- A. id
- B. class
- C. name
- D. placeholder

63. Какой input позволяет выбрать несколько независимых вариантов?

- A. checkbox
- B. radio
- C. submit
- D. email

64. Какой тег создаёт многострочное поле ввода?

- A. input type=text
- B. textarea
- C. select
- D. fieldset

65. Какой тег создаёт выпадающий список?

- A. select
- B. textarea
- C. legend
- D. label

66. Какой тег создаёт вариант внутри select?

- A. li
- B. option
- C. td
- D. button

67. Для чего нужен fieldset?

- A. Группировать связанные поля формы
- B. Создать HTTP-запрос
- C. Заменить table
- D. Подключить CSS

68. Для чего нужен legend?

- A. Заголовок группы fieldset
- B. Ссылка на CSS
- C. Скрытый input
- D. Порт HTTPS

69. Что делает placeholder?

- A. Задаёт подсказку внутри пустого поля
- B. Заменяет label полностью
- C. Отправляет форму
- D. Создаёт id

70. Какой type у button лучше явно указывать для отправки?

- A. button
- B. submit
- C. reset
- D. send

71. Какой input type скрыт от пользователя, но отправляется с формой?

- A. hidden
- B. password
- C. file
- D. radio

72. Почему серверная валидация всё равно нужна?

- A. HTML-валидацию можно обойти
- B. Браузер не умеет формы
- C. CSS запрещает required
- D. GET не отправляет данные

Блок 7. CSS selectors и каскад

73. Какой селектор выбирает все элементы?

- A. #all
- B. *
- C. .all
- D. all

74. Какой селектор выбирает элементы по классу?

- A. #card
- B. .card
- C. card
- D. [card]

75. Какой селектор выбирает элемент по id?

- A. .main
- B. #main
- C. main
- D. *main

76. Что выбирает div p?

- A. Только прямые p внутри div
- B. Все p внутри div на любой глубине
- C. Все div после p
- D. Только первый p

77. Что выбирает div > p?

- A. Все потомки p
- B. Только прямые дочерние p
- C. Все div
- D. Все p после div

78. Что выбирает h2 + p?

- A. Любой p внутри h2
- B. Первый p сразу после h2
- C. Все p перед h2
- D. Все h2 внутри p

79. Какой селектор ищет элемент по атрибуту required?

- A. input.required
- B. input#required
- C. input[required]
- D. input:required()

80. Какой псевдокласс срабатывает при наведении мыши?

- A. :focus
- B. :hover
- C. :active
- D. :checked

81. Какой псевдокласс выбирает второй элемент среди детей?

- A. :second
- B. :nth-child(2)
- C. ::second-child
- D. :child(2)

82. Что сильнее: id или class?

- A. id
- B. class
- C. Одинаково
- D. Зависит только от длины

83. Что сильнее: .content или div?

- A. .content
- B. div
- C. Одинаково
- D. Короткий селектор

84. Если специфичность одинаковая, какое правило победит?

- A. Первое в файле
- B. Последнее ниже по коду
- C. Более короткое
- D. С меньшим числом свойств

Блок 8. CSS box model и свойства

85. Из чего состоит CSS box model?

- A. content, padding, border, margin
- B. head, body, footer
- C. GET, POST, PUT
- D. model, view, template

86. Что такое padding?

- A. Внешний отступ
- B. Внутренний отступ
- C. Рамка
- D. Тень

87. Что такое margin?

- A. Внешний отступ
- B. Внутренний отступ
- C. Контент
- D. Порт

88. Что делает box-sizing: border-box?

- A. Убирает border
- B. Включает padding и border в width
- C. Делает inline
- D. Создаёт grid

89. Какая ширина у блока `width:100px; padding:10px; border:5px; box-sizing:border-box`?

- A. 100px
- B. 110px
- C. 120px
- D. 130px

90. Какое значение `display` делает элемент блочным?

- A. inline
- B. block
- C. none
- D. contents

91. Какое значение `display` убирает элемент из раскладки?

- A. hidden
- B. none
- C. block
- D. zero

92. Какое свойство задаёт семейство шрифта?

- A. font-family
- B. font-weight
- C. font-style
- D. line-height

93. Какое свойство задаёт толщину шрифта?

- A. font-size
- B. font-weight
- C. letter-spacing
- D. text-align

94. Какое свойство задаёт подчёркивание текста?

- A. text-align
- B. text-decoration
- C. text-shadow
- D. line-height

95. Какое свойство задаёт расстояние между строками?

- A. letter-spacing
- B. line-height
- C. font-style
- D. background-size

96. Какое свойство добавляет тень к тексту?

- A. box-shadow
- B. font-shadow
- C. text-shadow
- D. shadow-text

Блок 9. Flexbox и Grid

97. Как создать флекс-контейнер?

- A. display: block
- B. display: flex
- C. position: flex
- D. flex: container

98. Как создать строчный флекс-контейнер?

- A. display: inline-flex
- B. display: flex-inline
- C. display: inline-block-flex
- D. inline: flex

99. Что задаёт flex-direction?

- A. Направление главной оси
- B. Цвет фона
- C. Размер шрифта
- D. Количество строк grid

100. Что делает justify-content во Flexbox?

- A. Выравнивает по главной оси
- B. Выравнивает текст
- C. Создаёт порт
- D. Задаёт DNS

101. Что делает align-items во Flexbox?

- A. Выравнивает по поперечной оси
- B. Удаляет элемент
- C. Задаёт href
- D. Считает объекты ORM

102. Какое значение align-items по умолчанию?

- A. flex-start
- B. center
- C. stretch
- D. baseline

103. Что означает flex-grow?

- A. Коэффициент роста элемента
- B. Цвет элемента
- C. Порядок DNS
- D. HTTP-метод

104. Что означает flex-shrink?

- A. Коэффициент сжатия
- B. Высота строки
- C. Имя формы
- D. Тип модели

105. Как создать grid-контейнер?

- A. display: grid
- B. grid: true
- C. display: table-grid
- D. position: grid

106. Как называются линии, образующие структуру сетки?

- A. Grid Tracks
- B. Grid Cells
- C. Grid Areas
- D. Grid Lines

107. Что означает fr в Grid?

- A. Фиксированный пиксель
- B. Доля свободного пространства
- C. Форма регистрации
- D. Фрагмент URL

108. Что делает repeat(3, 1fr)?

- A. Создаёт три одинаковых трека
- B. Повторяет HTTP-запрос
- C. Создаёт три формы
- D. Выбирает третий элемент

109. Где обычно описываются модели Django?

- A. views.py
- B. models.py
- C. urls.py
- D. templates.py

110. Что описывает модель Django?

- A. HTML-шаблон
- B. Таблицу базы данных
- C. CSS-селектор
- D. DNS-зону

111. Что Django создаёт автоматически, если не указан primary_key?

- A. uuid
- B. pk_id
- C. id
- D. index

112. Какое поле требует max_length?

- A. TextField
- B. CharField
- C. BooleanField
- D. DateField

113. Что означает null=True?

- A. Можно хранить NULL в БД
- B. Можно оставить поле пустым в форме
- C. Поле становится id
- D. Поле становится CSS-классом

114. Что означает blank=True?

- A. Можно хранить NULL в БД
- B. Можно оставить пустым в форме/админке
- C. Поле уникально
- D. Поле удаляется

115. Что задаёт class Meta?

- A. Настройки модели
- B. HTTP-заголовки
- C. HTML-комментарии
- D. CSS-псевдоклассы

116. Какая связь описывает многие-к-одному?

- A. ManyToManyField
- B. ForeignKey
- C. OneToOneField
- D. CharField

117. Что делает on_delete=CASCADE?

- A. Запрещает удаление
- B. Удаляет зависимые объекты
- C. Ставит NULL
- D. Ничего не делает

118. Что задаёт related_name?

- A. Имя обратной связи
- B. Имя HTML-файла
- C. Имя URL-схемы
- D. Имя CSS-файла

119. Какой аргумент задаёт промежуточную модель ManyToMany?

- A. to_field
- B. intermediate
- C. through
- D. link_model

120. Каким методом можно добавить связь ManyToMany?

- A. append()
- B. add()
- C. insert()
- D. join()

Блок 11. Django ORM queries

121. Что такое ORM?

- A. Механизм работы с БД через объекты Python
- B. HTML-таблица
- C. CSS-фреймворк
- D. DNS-сервер

122. Что такое QuerySet?

- A. Ленивое представление запроса к БД
- B. HTTP-ответ
- C. HTML-форма
- D. CSS-блок

123. Что возвращает all()?

- A. Один объект
- B. QuerySet всех объектов
- C. Словарь агрегации
- D. Boolean

124. Что возвращает filter()?

- A. QuerySet
- B. Всегда один объект
- C. Только число
- D. HTML

125. Что возвращает get() при успешном поиске?

- A. QuerySet
- B. Один объект
- C. Список словарей
- D. CSS-правило

126. Что делает exclude()?

- A. Исключает объекты по условию
- B. Сортирует
- C. Удаляет таблицу
- D. Создаёт форму

127. Что возвращает aggregate()?

- A. Словарь с итоговыми значениями
- B. Каждый объект с новым полем
- C. HTML-шаблон
- D. URL

128. Что делает annotate()?

- A. Добавляет вычисляемое поле каждому объекту
- B. Удаляет дубли
- C. Проверяет пароль
- D. Создаёт cookie

129. Какой lookup ищет подстроку без учёта регистра?

- A. __exact
- B. __search
- C. __icontains
- D. __contains

130. Для чего нужны Q-объекты?

- A. Для OR/AND/NOT условий
- B. Для HTML-комментариев
- C. Для DNS-записей
- D. Для CSS Grid

131. Для чего нужны F-выражения?

- A. Работать со значениями полей на стороне БД
- B. Форматировать HTML
- C. Создавать URL
- D. Выбирать порт

132. Как работает select_related на уровне SQL?

- A. Делает SQL JOIN и получает данные за один запрос
- B. Группирует через GROUP BY
- C. Кэширует шаблоны
- D. Делает отдельный запрос для каждого объекта

Блок 12. Django URLs, views, templates и forms

133. В каком файле обычно настраиваются основные маршруты?

- A. models.py
- B. urls.py
- C. settings.py
- D. views.py

134. Какой аргумент `path()` позволяет обращаться к маршруту по имени?

- A. `alias`
- B. `id`
- C. `name`
- D. `ref`

135. Как правильно создать ссылку на маршрут `myapp:find` в шаблоне?

- A. `{% path "myapp:find" %}`
- B. `{% url "myapp:find" %}`
- C. `{{ search myapp.find }}`
- D. `<% url myapp:find %>`

136. Для чего используется `app_name`?

- A. Для namespace маршрутов приложения
- B. Для выбора БД
- C. Для CSS
- D. Для заголовка сайта

137. Что принимает `view` первым аргументом?

- A. `response`
- B. `request`
- C. `template`
- D. `model`

138. Что вернёт шаблон, если переменной нет в `context`?

- A. `NameError`
- B. `None`
- C. Ошибка 500 всегда
- D. Пустую строку

139. Какой синтаксис у шаблонных тегов Django?

- A. {% tag %}
- B. {{ tag }}
- C. {# tag #}
- D. <% tag %>

140. Как обратиться к user_data["name"] в шаблоне?

- A. {{ user_data.name }}
- B. {{ user_data["name"] }}
- C. {{ user_data.get("name") }}
- D. {{ user_data→name }}

141. Что будет, если в ListView не указать model и queryset?

- A. Django сам найдёт модель
- B. Будет ошибка
- C. Покажет все объекты
- D. Покажет пустой список

142. Какой метод CBV добавляет context?

- A. get_template_context()
- B. set_context_data()
- C. get_context_data()
- D. get_context()

143. Какой метод формы проверяет связь между полями?

- A. save()
- B. validate_fields()
- C. clean_password()
- D. clean()

144. Что появляется после успешного form.is_valid()?

- A. cleaned_data
- B. urlpatterns
- C. box_model
- D. DNS cache

Ключ ответов

1 — B	2 — B	3 — D	4 — C	5 — B	6 — B	7 — C	8 — B	9 — A	10 — B	11 — C	12 — D
13 — B	14 — B	15 — C	16 — A	17 — C	18 — A	19 — B	20 — D	21 — A	22 — B	23 — A	24 — B
25 — A	26 — C	27 — B	28 — B	29 — C	30 — D	31 — B	32 — B	33 — A	34 — B	35 — A	36 — B
37 — C	38 — B	39 — B	40 — A	41 — C	42 — A	43 — A	44 — A	45 — B	46 — A	47 — B	48 — B
49 — B	50 — A	51 — A	52 — A	53 — B	54 — C	55 — B	56 — C	57 — B	58 — B	59 — B	60 — B
61 — B	62 — C	63 — A	64 — B	65 — A	66 — B	67 — A	68 — A	69 — A	70 — B	71 — A	72 — A
73 — B	74 — B	75 — B	76 — B	77 — B	78 — B	79 — C	80 — B	81 — B	82 — A	83 — A	84 — B
85 — A	86 — B	87 — A	88 — B	89 — A	90 — B	91 — B	92 — A	93 — B	94 — B	95 — B	96 — C
97 — B	98 — A	99 — A	100 — A	101 — A	102 — C	103 — A	104 — A	105 — A	106 — D	107 — B	108 — A
109 — B	110 — B	111 — C	112 — B	113 — A	114 — B	115 — A	116 — B	117 — B	118 — A	119 — C	120 — B
121 — A	122 — A	123 — B	124 — A	125 — B	126 — A	127 — A	128 — A	129 — C	130 — A	131 — A	132 — A

133 — B	134 — C	135 — B	136 — A	137 — B	138 — D	139 — A	140 — A	141 — B	142 — C	143 — D	144 — A
-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------	-------------------