

Web-программирование

Тест №8

(2 час)

Кислов С.В.

09.03.01.

ИВМ-3, 3 курс

Мест № 8

Местовый контроль "Основы создания библиотек Composer"

Часть 1. Размещение пакета в репозитории пакетов Composer и Команды Git для добавления пакетов;

Вопросы с выбором;

1. а. Это публичный репозиторий пакетов Composer
2. а. Зарегистрироваться на сайте Packagist и добавить URL ~~свое~~ своего репозитория;
3. а. Создать репозиторий на GitHub или другом хостинге, затем переименовать его подключить его к Packagist;
4. а. Git репозиторий (например, GitHub);

5. d. Все поля name, description, version, autoload, license.

6. a. Выпустить новый мер в вашем репозитории Bit и отправить изменения на Packagist;

7. a. Использовать параметр repositories в файле composer.json;

8. a. Пакет не будет доступен для установки;

9. b. Название, описание, версия, авторы, лицензия, зависимости, автозагрузка;

10. a. composer validate
Вопрос по команде Bit для добавления пакетов и публикации на Packagist;

12. b. git init

~~13. a. git push~~

~~14.~~

13. a. git add

13. b. git push

14. a. git tag -a v1.0.0

~~15.~~

15. a. git push origin v1.0.0

16. c. git tag

17. b. git tag -d <tag-name>

18. d. git pull

19. c. git checkout -b <branch-name>

20. b. git checkout <branch-name>

Вопросы на сопоставление

1 E

2 B

3 C

4 D

5 A

21.-30. Имперские восходы

21. Имберн 21

22. Имберн 22

23. Имберн 23

24. Имберн 24

25. Имберн 25

26. Имберн 26

27. Имберн 27

28. Имберн 28

29.

Imberr 29

30.

Imberr 30

Ред

Месон №8

Куров Г.В.

3 куче

(2 уакоб)

21. Как зарегистрировать свой пакет на Packagist и какие шаги нужно выполнить для его размещения?

- Чтобы зарегистрировать пакет на Packagist, необходимо выполнить следующие шаги:
 1. Создать учётную запись на Packagist.
 2. Создать репозиторий на GitHub (или другом сервисе) и добавить в него файл `composer.json`.
 3. Убедиться, что в репозитории есть теги версий (например, `v1.0.0`).
 4. Перейти на Packagist, нажать "Submit" и ввести URL вашего репозитория.
 5. Packagist автоматически проверит репозиторий и добавит его в свой список пакетов.

Ответ 21

22. Как настроить репозиторий на GitHub (или другом сервисе) для корректной работы с Packagist?

- Для корректной работы с Packagist необходимо:
 1. Убедиться, что репозиторий публичный (или настроить доступ для Packagist, если он частный).
 2. Включить в репозиторий файл `composer.json` с корректными данными.
 3. Создать теги версий для релизов (например, `v1.0.0`).

Ответ 22

23. Какие требования предъявляются к версии пакета, чтобы он корректно отображался в репозитории Packagist?

- Версия пакета должна соответствовать формату семантического версионирования (например, 1.0.0).

Версия должна быть указана в теге Git (например, v1.0.0).

Version 23

24. Как обновить пакет на Packagist после того, как новая версия была выпущена в вашем репозитории? Опишите весь процесс.

- Для обновления пакета на Packagist:
 1. Создайте новый тег версии в репозитории (например, v1.1.0).
 2. Packagist автоматически обнаружит новый тег и обновит информацию о пакете.

Ответ 24

25. Как добавить свой собственный частный репозиторий пакетов в проект с помощью Composer? Какие параметры для этого необходимо указать в composer.json?

- Чтобы добавить частный репозиторий, в composer.json нужно указать:

- {
- "repositories": [- {
- "type": "vcs",
- "url": "https://github.com/username/repository"
- }
-]

}

- Если репозиторий частный, можно добавить параметры аутентификации в auth.json.

Domven 25

26. Какие проблемы могут возникнуть, если в репозитории пакета отсутствуют теги Git, и как это можно исправить?

- Без тегов Git Packagist не сможет определить версии пакета.
- Для исправления нужно создать теги версий (например, `git tag v1.0.0` и `git push origin v1.0.0`).

Ответ 26)

27. Какие поля должны быть обязательно указаны в файле `composer.json`, чтобы пакет корректно отображался на Packagist?

- Обязательные поля: name, description, version, type, license, authors.

Ответ 24

28. Что нужно учитывать при управлении зависимостями для пакетов, размещённых на частных репозиториях в Composer?

- Убедитесь, что у вас есть доступ к частному репозиторию.
- Настройте аутентификацию в `auth.json` для доступа к частным репозиториям.

Ответ 28

29. Опишите процесс публикации нового пакета в репозитории Composer, начиная с его создания до публикации на Packagist.

- Создайте репозиторий на GitHub и добавьте composer.json.
- Создайте теги версий.
- Зарегистрируйтесь на Packagist и добавьте URL вашего репозитория.
- Packagist автоматически добавит ваш пакет в свой список.

Answer 29

30. Как Composer взаимодействует с другими репозиториями, если пакет не был опубликован на Packagist? Как указать дополнительные репозитории для установки зависимостей?

- Composer может взаимодействовать с другими репозиториями, указав их в composer.json:

```
{  
  "repositories": [  
    {  
      "type": "vcs",  
      "url": "https://github.com/username/repository"  
    }  
  ]  
}
```

- Это позволяет Composer находить и устанавливать пакеты из указанных репозиториях.

Дмбевн 30