GIT Homework 1

JSON

1. Создать внешний репозиторий c названием JSON.

1) Заходим в свой профиль на Git Hub.

2) Нажимаем "Repositories"

3) Далее кнопка "New"

4) Пишем название репозитория "JSON", ставим галочку напротив поля "Add a README file", нажимаем на кнопку "Create repository"

2. Клонировать репозиторий JSON на локальный компьютер.

1) Заходим в созданный репозиторий JSON

2) Нажимаем на кнопку "Code"

3) Копируем ссылку

4) Заходим в Git Bash

5) Пишем команду

git clone скопированная ссылка с Git Hub

3. Внутри локального JSON создать файл “new.json”.

1) cd JSON - заходим в папку JSON

2) touch new.json или cat > new.json - создаем файл

4. Добавить файл под гит.

git add new.json

5. Закоммитить файл.

git commit -m "add 1 file"

6. Отправить файл на внешний GitHub репозиторий.

git push

7. Отредактировать содержание файла “new.json” - написать информацию о себе (ФИО, возраст, количество домашних животных, будущая желаемая зарплата). Всё написать в формате JSON.

1) vim new.json - открываем редактор файла

2) i - начать редактирование

3) { "surname": "Ryzhkovich", "name": "Valeria", "patronymic\_name": "Sergeevna", "pets": 0, "preferred\_salary": "500 dollars"}

4) :wq - сохраняем и выходим из редактора

8. Отправить изменения на внешний репозиторий.

1) git add new.json - добавляем файл под гит

2) git commit -m "add changes new.json" - закоммитили изменения

3) git push - отправляем на внешний репозиторий

9. Создать файл preferences.json

touch preferences.json или cat > preferences.json

10. В файл preferences.json добавить информацию о своих предпочтениях (Любимый фильм, любимый сериал, любимая еда, любимое время года, сторона которую хотели бы посетить) в формате JSON.

1) vim preferences.json - открываем редактор файла

2) i - начать редактирование

3) { "favorite\_movie": "Knocking on heaven's door", "favorite\_show": "The witcher", "favorite\_food": "Italian", "favorite\_season": "Summer", "country": "Spain"}

4) :wq - сохраняем и выходим из редактора

11. Создать файл sklls.json добавить информацию о скиллах которые будут изучены на курсе в формате JSON

1) vim sklls.json - открываем редактор файла

2) i - начать редактирование

3) { "soft\_skills":

{ "1": "Communication skills",

"2": "Analytical skills",

"3": "Attention to detail",

"4": "Teamwork",

"5": "Focus on quality",

"6": "Patient work"},

"hard\_skills":

{ "1": "Knowledge of software testing types, methodologies, and techniques",

"2": "Familiarity with certain bug tracking and test management tools",

"3": "Mobile testing",

"4": "Knowledge of a particular programming language"}

}

4) :wq - сохраняем и выходим из редактора

12. Отправить сразу 2 файла на внешний репозиторий.

1) git add . - добавлем все под гит

2) git status - смотрим, отслеживает ли гит наши файлы

2) git commit -m "add new files" - закоммитили изменения

3) git push - отправляем на внешний репозиторий

13. На веб интерфейсе создать файл bug\_report.json.

1) В репозитории нажимаем кнопку "Add file" -> "Create new file"

2) В поле "Name your file ..." пишем название bug\_report.json

14. Сделать Commit changes (сохранить) изменения на веб интерфейсе.

Нажимаем на кнопку "Commit new file"

15. На веб интерфейсе модифицировать файл bug\_report.json, добавить баг репорт в формате JSON.

{"fields":

{"ID": "1",

"Title": "Оплата товара",

"Priority": "high",

"Severity": "Major",

"Environment": "Samsung Galaxy",

"Steps": "{'1': 'Зайти на сайт', '2': 'Найти товар', '3': 'Добавить в корзину', '4': 'Оплатить товар'}",

"ExpectedResult": "оплата прошла",

"ActualResult":"оплата не прошла"}

}

16. Сделать Commit changes (сохранить) изменения на веб интерфейсе.

Нажимаем на кнопку "Commit changes"

17. Синхронизировать внешний и локальный репозиторий JSON

git pull

XML

1. Создать внешний репозиторий c названием XML.

1) Заходим в свой профиль на Git Hub.

2) Нажимаем "Repositories"

3) Далее кнопка "New"

4) Пишем название репозитория "XML", ставим галочку напротив поля "Add a README file", нажимаем на кнопку "Create repository"

2. Клонировать репозиторий XML на локальный компьютер.

1) Заходим в созданный репозиторий XML

2) Нажимаем на кнопку "Code"

3) Копируем ссылку

4) Заходим в Git Bash

5) Пишем команду

git clone скопированная ссылка с Git Hub

3. Внутри локального XML создать файл “new.xml”.

1) cd XML - заходим в папку XML

2) touch new.xml или cat > new.xml - создаем файл

4. Добавить файл под гит.

git add new.xml

5. Закоммитить файл.

git commit -m "add 1 file"

6. Отправить файл на внешний GitHub репозиторий.

git push

7. Отредактировать содержание файла “new.xml” - написать информацию о себе (ФИО, возраст, количество домашних животных, будущая желаемая зарплата). Всё написать в формате XML.

1) vim preferences.xml - открываем редактор файла

2) i - начать редактирование

3) <?xml version="1.0" encoding="windows-1251"?>

<root>

<surname>Ryzhkovich</surname>

<name>Valeria</name>

<patronymic\_name>Sergeevna</patronymic\_name>

<pets>0</pets>

<preferred\_salary>500 dollars</preferred\_salary>

</root>

4) :wq - сохраняем и выходим из редактора

8. Отправить изменения на внешний репозиторий.

1)git commit -am "add changes new.xml" - закоммитили файл

2) git push - отправили на внешний репозиторий

9. Создать файл preferences.xml

touch preferences.xml

10. В файл preferences.xml добавить информацию о своих предпочтениях (Любимый фильм, любимый сериал, любимая еда, любимое время года, сторона которую хотели бы посетить) в формате XML.

1) vim preferences.xml - открываем редактор файла

2) i - начать редактирование

3)<root>

<favorite\_movie>Knocking on heaven's door</favorite\_movie>

<favorite\_show>The witcher</favorite\_show>

<favorite\_food>Italian</favorite\_food>

<favorite\_season>Summer</favorite\_season>

<country>Spain</country>

</root>

4):wq - сохраняем и выходим из редактора

11. Создать файл sklls.xml добавить информацию о скиллах которые будут изучены на курсе в формате XML

1) vim sklls.xml - открываем редактор файла

2) i - начать редактирование

3) <root>

<soft\_skills>

<item1>Communication skills</item1>

<item2>Analytical skills</item2>

<item3>Attention to detail</item3>

<item4>Teamwork</item4>

<item5>Focus on quality</item5>

<item6>Patient work</item6>

</soft\_skills>

<hard\_skills>

<point1>Knowledge of software testing types, methodologies, and techniques</point1>

<point2>Familiarity with certain bug tracking and test management tools</point2>

<point3>Mobile testing</point3>

<point4>Knowledge of a particular programming language</point4>

</hard\_skills>

</root>

4):wq - сохраняем и выходим из редактора

12. Сделать коммит в одну строку.

git add . | git commit -m "add new files"

13. Отправить сразу 2 файла на внешний репозиторий.

git push

14. На веб интерфейсе создать файл bug\_report.xml.

1) В репозитории нажимаем кнопку "Add file" -> "Create new file"

2) В поле "Name your file ..." пишем название bug\_report.xml

15. Сделать Commit changes (сохранить) изменения на веб интерфейсе.

Нажимаем на кнопку "Commit new file"

16. На веб интерфейсе модифицировать файл bug\_report.xml, добавить баг репорт в формате XML.

<?xml version="1.0" encoding="UTF-8" ?>

<fields>

<ID>1</ID>

<Title>Оплата товара</Title>

<Priority>high</Priority>

<Severity>Major</Severity>

<Environment>Samsung Galaxy</Environment>

<Steps>{'1': 'Зайти на сайт', '2': 'Найти товар', '3': 'Добавить в корзину', '4': 'Оплатить товар'}</Steps>

<ExpectedResult>оплата прошла</ExpectedResult>

<ActualResult>оплата не прошла</ActualResult>

</fields>

17. Сделать Commit changes (сохранить) изменения на веб интерфейсе.

Нажимаем на кнопку "Commit changes"

18. Синхронизировать внешний и локальный репозиторий XML

git pull

TXT

1. Создать внешний репозиторий c названием TXT.

1) Заходим в свой профиль на Git Hub.

2) Нажимаем "Repositories"

3) Далее кнопка "New"

4) Пишем название репозитория " TXT ", ставим галочку напротив поля "Add a README file", нажимаем на кнопку "Create repository"

2. Клонировать репозиторий TXT на локальный компьютер.

1) Заходим в созданный репозиторий TXT

2) Нажимаем на кнопку "Code"

3) Копируем ссылку

4) Заходим в Git Bash

5) Пишем команду

git clone скопированная ссылка с Git Hub

3. Внутри локального TXT создать файл “new.txt”.

1) cd TXT - заходим в папку TXT

2) vim new.txt или cat > new.txt - создаем файл

4. Добавить файл под гит.

git add new.txt

5. Закоммитить файл.

git commit -m "add 1 file"

6. Отправить файл на внешний GitHub репозиторий.

git push

7. Отредактировать содержание файла “new.txt” - написать информацию о себе (ФИО, возраст, количество домашних животных, будущая желаемая зарплата). Всё написать в формате TXT.

1) vim new.txt - открываем редактор файла

2) i - начать редактирование

3) Ryzhkovich Valeria Sergeevna. I haven't got a pet. My preferred salary is 500 dollars.

4) :wq - сохраняем и выходим из редактора

8. Отправить изменения на внешний репозиторий.

1) git commit -am "add changes new.txt " - закоммитили изменения

2) git push - отправляем на внешний репозиторий

9. Создать файл preferences.txt

touch preferences.txt

10. В файл preferences.txt” добавить информацию о своих предпочтениях (Любимый фильм, любимый сериал, любимая еда, любимое время года, сторона которую хотели бы посетить) в формате TXT.

1) vim preferences.txt - открываем редактор файла

2) i - начать редактирование

3) My favorite movie is knocking on heaven's door. My favorite show is the witcher. My favorite food is Italian. My favorite season is summer. Country which I want to visit is Spain.

4) :wq - сохраняем и выходим из редактора

11. Создать файл sklls.txt добавить информацию о скиллах которые будут изучены на курсе в формате TXT

1) vim sklls.txt

2) i - начать редактирование

3)The soft skills are communication skills, analytical skills, attention to detail, teamwork, focus on quality, patient work.

The hard skills are:

1. Knowledge of software testing types, methodologies, and techniques.

2. Familiarity with certain bug tracking and test management tools.

3. Mobile testing.

4. Knowledge of a particular programming language.

4) :wq - сохраняем и выходим из редактора

12. Сделать коммит в одну строку.

git commit -am "add new files"

13. Отправить сразу 2 файла на внешний репозиторий.

git push

14. На веб интерфейсе создать файл bug\_report.txt.

1) В репозитории нажимаем кнопку "Add file" -> "Create new file"

2) В поле "Name your file ..." пишем название bug\_report.txt

15. Сделать Commit changes (сохранить) изменения на веб интерфейсе.

Нажимаем на кнопку "Commit new file"

16. На веб интерфейсе модифицировать файл bug\_report.txt, добавить баг репорт в формате TXT.

A bug report should contain the following fields:

1)ID: 1

2)Title: Оплата товара

3)Priority: high

4)Severity: Major

5)Environment: Samsung Galaxy

6)Steps:

1: Зайти на сайт

2: Найти товар

3: Добавить в корзину

4: Оплатить товар

7)Expected Result: оплата прошла

8)Actual Result: оплата не прошла

17. Сделать Commit changes (сохранить) изменения на веб интерфейсе.

Нажимаем на кнопку "Commit changes"

18. Синхронизировать внешний и локальный репозиторий TXT

git pull