|  |  |  |
| --- | --- | --- |
| **GIT. HOMEWORK\_1** | | |
| **№** | **Do** | **Done** |
| **0.** | **Аккаунт в GitHub** | **https://github.com/raskote** |
| **JSON** | | |
|  | Создать внешний репозиторий c названием JSON. | **Repository created** |
|  | Клонировать репозиторий JSON на локальный компьютер. | git clone git@github.com:raskote/JSON.git |
|  | Внутри локального JSON создать файл “new.json”. | cd JSON  touch new.json |
|  | Добавить файл под гит. | git add . |
|  | Закоммитить файл. | git commit -m ‘added new.json’ |
|  | Отправить файл на внешний GitHub репозиторий. | git push |
|  | Отредактировать содержание файла “new.json” - написать информацию о себе (ФИО, возраст, количество домашних животных, будущая желаемая зарплата). Всё написать в формате JSON. | nano new.json  {  “name” : “Bogdanov Mikhail Borisovich”,  “age” : 32,  “count of pets” : 1,  “required payment” : 100000  } |
|  | Отправить изменения на внешний репозиторий. | git add .  git commit -m “change of new.json”  git push |
|  | Создать файл preferences.json | touch preferences.json |
|  | 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. | nano preferences.json  {  “favourite movie” : “the dark knight”,  “favourite show” : “the friends”,  “favourite food” : “kfc”,  “favourite season of year” : “summer”,  “country which I want to visit” : “Germany” } |
|  | Создать файл sklls.json добавить информацию о скиллах которые будут изучены на курсе в формате JSON | touch skills.json  nano skills.json  {  "skill\_0" : "Базовая теория",  "skill\_1" : "Что такое клиент-серверная архитектура",  "skill\_2" : "HTTP методы запросов на сервер",  "skill\_3" : "Коды ответов HTTP сервера",  "skill\_4" : "Структуры HTTP запросов и ответов",  "skill\_5" : "Что такое JSON, XML. Их структура",  "skill\_6" : "Тестирование API через Postman (JS, автотесты API)",  "skill\_7" : "Снятие и чтение логов с внешнего сервера",  "skill\_8" : "Снифинг http web трафика через Charles и Fiddler",  "skill\_9" : "Dev Tools веб браузеров (Google Chrome, FireFox)",  "skill\_10" : "VPN. (Как работает, зачем нужен, как использовать, варианты инструментов)",  "skill\_11" : "Мобильное тестирование",  "skill\_12" : "Особенность iOS, Android, гайдлайны",  "skill\_13" : "Сборка iOS приложений на XCode",  "skill\_14" : "Сборка Android приложений на Android Studio",  "skill\_15" : "ADB (управление андройд девайсами)",  "skill\_16" : "Настройка прокси и vpn на iOS и Android",  "skill\_17" : "Перехват (сниффинг) мобильного трафика через Charles и Fiddler на iOS и Android",  "skill\_18" : "Командная строка (terminal) Linux (копирование, создание, просмотр, перемещение файлов на серверах без графического интерфейса)",  "skill\_19" : "Основы bash скриптинг, автоматизация рутинных задач на сервере",  "skill\_20" : "Доступ к удалённым серверам",  "skill\_21" : "Основы SQL (Create, Delete, Drop, Insert Into, Select, From, Where, Join)",  "skill\_22" : "База данных Postgres (установка, настройка и использование)",  "skill\_23" : "Нереляционная база данных Redis (установка, настройка и использование)",  "skill\_24" : "Нагрузочное тестирование в Jmeter",  "skill\_25" : "Методология разработки Scrum",  "skill\_26" : "Python. (Изучение основ. Создание клиент серверного приложения)"  } |
|  | Отправить сразу 2 файла на внешний репозиторий. | git add .  git commit -m “added 2 json files”  git push |
|  | На веб интерфейсе создать файл bug\_report.json. | add file - create new file  bug\_report.json |
|  | Сделать Commit changes (сохранить) изменения на веб интерфейсе. | commit new file |
|  | На веб интерфейсе модифицировать файл bug\_report.json, добавить баг репорт в формате JSON. | {  "Title" : "Example",  "Description" : "Example",  "Preposition" : "Example",  "Steps to reproduce" : "Example",  "Expected result" : "Example",  "Actual result" : "Example",  "Attachments": "Example"  } |
|  | Сделать Commit changes (сохранить) изменения на веб интерфейсе. | commit changes |
|  | Синхронизировать внешний и локальный репозиторий JSON | git pull |
| **XML** | | |
|  | Создать внешний репозиторий c названием XML. | Repository created |
|  | Клонировать репозиторий XML на локальный компьютер. | git clone git@github.com:raskote/XML.git |
|  | Внутри локального XML создать файл “new.xml”. | touch new.xml |
|  | Добавить файл под гит. | git add . |
|  | Закоммитить файл. | git commit -m “added new.xml” |
|  | Отправить файл на внешний GitHub репозиторий. | git push |
|  | Отредактировать содержание файла “new.xml” - написать информацию о себе (ФИО, возраст, количество домашних животных, будущая желаемая зарплата). Всё написать в формате XML. | nano new.xml  <?xml version="1.0" standalone="yes" ?>  - <NewDataSet>  - <Table>  <ФИО>Богданов Михаил Борисович</ФИО>  <Возраст>32</Возраст>  <Количество\_животных>1</Количество\_животных>  <Желаемая\_зарплата>100000</Желаемая\_зарплата>  </Table> |
|  | Отправить изменения на внешний репозиторий. | git add .  git commit –m “change of new.xml”  git push |
|  | Создать файл preferences.xml | touch preferences.xml |
|  | 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. | Nano preferences.xml  <?xml version="1.0" standalone="yes" ?>  <NewDataSet>  <Table>  <Любимый\_фильм>Темный рыцарь</Любимый\_фильм>  <Любимый\_сериал>Друзья</Любимый\_сериал>  <Любимая\_еда>KFC</Любимая\_еда>  <Любимое\_время\_года>Лето</Любимое\_время\_года>  <Страна\_которую\_я\_хотел\_бы\_посетить>Германия<Страна\_которую\_я\_хотел\_бы\_посетить>  </Table> |
|  | Создать файл sklls.xml добавить информацию о скиллах которые будут изучены на курсе в формате XML | touch skills.xml  nano skill.xml  <?xml version=>1.0> standalone=>yes> ?>  <NewDataSet>  <Table>  <skill\_0>Базовая теория</skill\_0>  <skill\_1>Что такое клиент-серверная архитектура</skill\_1>  <skill\_2>HTTP методы запросов на сервер</skill\_2>  <skill\_3>Коды ответов HTTP сервера</skill\_3>  <skill\_4>Структуры HTTP запросов и ответов</skill\_4>  <skill\_5>Что такое JSON XML. Их структура</skill\_5>  <skill\_6>Тестирование API через Postman (JS автотесты API)</skill\_6>  <skill\_7>Снятие и чтение логов с внешнего сервера</skill\_7>  <skill\_8>Снифинг http web трафика через Charles и Fiddler</skill\_8>  <skill\_9>Dev Tools веб браузеров (Google Chrome FireFox)</skill\_9>  <skill\_10>VPN. (Как работает зачем нужен как использовать варианты инструментов)</skill\_10>  <skill\_11>Мобильное тестирование</skill\_11>  <skill\_12>Особенность iOS Android гайдлайны</skill\_12>  <skill\_13>Сборка iOS приложений на XCode</skill\_13>  <skill\_14>Сборка Android приложений на Android Studio</skill\_14>  <skill\_15>ADB (управление андройд девайсами)</skill\_15>  <skill\_16>Настройка прокси и vpn на iOS и Android</skill\_16>  <skill\_17>Перехват (сниффинг) мобильного трафика через Charles и Fiddler на iOS и Android</skill\_17>  <skill\_18>Командная строка (terminal) Linux (копирование создание просмотр перемещение файлов на серверах без графического интерфейса)<skill\_18>  <skill\_19>Основы bash скриптинг автоматизация рутинных задач на сервере</skill\_19>  <skill\_20>Доступ к удалённым серверам</skill\_20>  <skill\_21>Основы SQL (Create Delete Drop Insert Into Select From Where Join)</skill\_21>  <skill\_22>База данных Postgres (установка настройка и использование)</skill\_22>  <skill\_23>Нереляционная база данных Redis (установка настройка и использование)</skill\_23>  <skill\_24>Нагрузочное тестирование в Jmeter</skill\_24>  <skill\_25>Методология разработки Scrum</skill\_25>  <skill\_26>Python. (Изучение основ. Создание клиент серверного приложения)</skill\_26>  </Table> |
|  | Сделать коммит в одну строку. | git add .  git commit -m “change of preferences.xml” |
|  | Отправить сразу 2 файла на внешний репозиторий. | git push |
|  | На веб интерфейсе создать файл bug\_report.xml. | add file - create new file |
|  | Сделать Commit changes (сохранить) изменения на веб интерфейсе. | commit new file |
|  | На веб интерфейсе модифицировать файл bug\_report.xml, добавить баг репорт в формате XML. | <?xml version="1.0" standalone="yes" ?>  <NewDataSet>  <Table>  <Title>Example</Title>>  <Description>Example</Description>  <Preposition>Example</Preposition>  <Steps\_to\_reproduce>Example</Steps\_to\_reproduce>  <Expected\_result>Example</Expected\_result>  <Actual\_result>Example</Actual\_result>  <Attachments>Example</Attachments>  </Table> |
|  | Сделать Commit changes (сохранить) изменения на веб интерфейсе. | commit changes |
|  | Синхронизировать внешний и локальный репозиторий XML | git pull |
| **TXT** | | |
|  | Создать внешний репозиторий c названием TXT. | Repository created |
|  | Клонировать репозиторий TXT на локальный компьютер. | git clone git@github.com:raskote/TXT.git |
|  | Внутри локального TXT создать файл “new.txt”. | touch new.txt |
|  | Добавить файл под гит. | git add . |
|  | Закоммитить файл. | git commit -m “added new.txt” |
|  | Отправить файл на внешний GitHub репозиторий. | git push |
|  | Отредактировать содержание файла “new.txt” - написать информацию о себе (ФИО, возраст, количество домашних животных, будущая желаемая зарплата). Всё написать в формате TXT. | nano new.txt  ФИО: Богданов Михаил Борисович  Возраст: 32  Количество животных: 1  Желаемая\_зарплата: 100000 |
|  | Отправить изменения на внешний репозиторий. | git add .  git commit -m “change of new.txt”  git pull |
|  | Создать файл preferences.txt | touch preferences.txt |
|  | 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. | nano preferences.txt  Любимый фильм: Темный рыцарь  Любимый сериал: Друзья  Любимая еда: KFC  Любимое время года: Лето  Страна которую я хотел бы посетить: Германия |
|  | Создать файл sklls.txt добавить информацию о скиллах которые будут изучены на курсе в формате TXT | touch skills.txt  nano skills.txt  Список изучаемых навыков на курсе:  Базовая теория  Что такое клиент-серверная архитектура  HTTP методы запросов на сервер  Коды ответов HTTP сервера  Структуры HTTP запросов и ответов  Что такое JSON XML. Их структура  Тестирование API через Postman (JS автотесты API)  Снятие и чтение логов с внешнего сервера  Снифинг http web трафика через Charles и Fiddler  Dev Tools веб браузеров (Google Chrome FireFox)  VPN. (Как работает зачем нужен как использовать варианты инструментов)  Мобильное тестирование  Особенность iOS Android гайдлайны  Сборка iOS приложений на XCode  Сборка Android приложений на Android Studio  ADB (управление андройд девайсами)  Настройка прокси и vpn на iOS и Android  Перехват (сниффинг) мобильного трафика через Charles и Fiddler на iOS и Android  Командная строка (terminal) Linux (копирование создание просмотр перемещение файлов на серверах без графического интерфейса)<skill\_18>  Основы bash скриптинг автоматизация рутинных задач на сервере  Доступ к удалённым серверам  Основы SQL (Create Delete Drop Insert Into Select From Where Join)  База данных Postgres (установка настройка и использование)  Нереляционная база данных Redis (установка настройка и использование)  Нагрузочное тестирование в Jmeter  Методология разработки Scrum  Python. (Изучение основ. Создание клиент серверного приложения) |
|  | Сделать коммит в одну строку. | git add .  git commit -m “added 2 txt files” |
|  | Отправить сразу 2 файла на внешний репозиторий. | git push |
|  | На веб интерфейсе создать файл bug\_report.txt | add file - create new file |
|  | Сделать Commit changes (сохранить) изменения на веб интерфейсе. | commit new file |
|  | На веб интерфейсе модифицировать файл bug\_report.txt, добавить баг репорт в формате TXT. | Title: Example  Description: Example  Preposition: Example  Steps to reproduce: Example  Expected result: Example  Actual result: Example  Attachments: Example |
|  | Сделать Commit changes (сохранить) изменения на веб интерфейсе. | Commit changes |
|  | Синхронизировать внешний и локальный репозиторий TXT | git pull |

|  |  |  |
| --- | --- | --- |
| **GIT. HOMEWORK\_2** | | |
|  | На локальном репозитории сделать ветки для: | mkdir HW\_2  cd HW\_2  git init  echo “kek\_cheburek” > start.txt  git add .  git commit -m “added start.txt” |
| 1.1 | - Postman | git branch Postman |
| 1.2 | - Jmeter | git branch Jmeter |
| 1.3 | - CheckLists | git branch CheckLists |
| 1.4 | - Bag Reports | git branch Bag Reports |
| 1.5 | - SQL | git branch SQL |
| 1.6 | - Charles | git branch Charles |
| 1.7 | - Mobile testing | git branch Mobile\_testing |
|  | Запушить все ветки на внешний репозиторий | git remote add origin git@github.com:raskote/HW\_2.git  git branch -M main  git push origin --all |
|  | В ветке Bag Reports сделать текстовый документ со структурой баг репорта | git checkout Bag\_Reports  touch structurebr.txt  nano structurebr.txt  Title: Example  Description: Example  Preposition: Example  Steps to reproduce: Example  Expected result: Example  Actual result: Example  Attachments: Example |
|  | Запушить структуру багрепорта на внешний репозиторий | git add .  git commit -m “added structurebr.txt”  git push |
|  | Вмержить ветку Bag Reports в Main | git checkout main  git merge Bag\_Reports |
|  | Запушить main на внешний репозиторий. | git push |
|  | В ветке CheckLists набросать структуру чек листа. | git checkout CheckLists  touch structurecl.txt  nano structurecd.txt  ID: Example  Title: Example  Steps: Example  Expected Result: Example  Actual Result: Example |
|  | Запушить структуру на внешний репозиторий | git add .  git commit -m “added structurecl.txt”  git push |
|  | На внешнем репозитории сделать Pull Request ветки CheckLists в main | compare & pull request  merge |
|  | Синхронизировать Внешнюю и Локальную ветки Main | git checkout main  git pull |