**Четвертое занятие**

**Начинаем работать с ветками в GIT**

В процессе разработки вашего проекта на Github вы можете создавать организации на Гитхабе, команды, сможете создавать репозитории, которые будут связаны только с определенными командами и организациями. Но прежде чем дорасти до такого уровня, нам нужно посмотреть, как работать с репозиториями, если вы к примеру хотите доделать домашку, работу, или просто хотите кому-то помочь с open source проектом. Для этой цели на github существует pull request и fork?

Что такое fork репозитория? Fork чьего -то репозитория – это полное копирование его содержимого на свой аккаунт. Это нужно, чтобы получить «копию», которую можно использовать в собственных целях как угодно – к примеру, дописать проект в том виде, как вы считаете нужным. Так тоже бывает, но чаще всего fork делают с целью того, чтобы совершить pull request – «предложение по изменениям».

Обычно pull request выглядит следующим образом:

- Вы делаете fork репозитория, которому хотите помочь

- Скачиваете его к себе на компьютер, редактируете таким образом, как считаете нужным

- После этого сохраняете все изменения в другую ветку, к примеру, в ветку с именем «help”

- Пушите это в свой репозиторий, и после этого можете предложить pull request с последними автору оригинального репозитория. Изменения не записываются, понятное дело, автоматически, они висят в виде предложения. После этого автор репозитория, или тот человек, который занимается review, оценивает, насколько ваши изменения подходят, и если все ок, тогда происходит merge. Если изменения не подходят, тогда pull request можно отклонить.

Кортежи.

Кортеж выглядит как список, не считая того, что вместо квадратных скобок он использует круглые. После инициализации кортежа вы можете обращаться к его индексам точно таким же образом, как это делаете со списком.

Кортеж нужно использовать, если вы нуждаетесь в неизменяемости типов данных.

**Замена кортежа**

Элементы кортежа не могут изменяться, но вы можете изменить только все элементы зараз.

<code>

Dim = (100, 200)

# этот код не вызовет ошибки, потому что мы не меняем целиком все значения

Dim = (300, 400)

Dim[0] = 800

# а вот это уже не работает, потому что мы пытаемся отредактировать только отдельный элемент

</code>

Кортежи и множества

Важные замечания:

Если Вы совершаете fork репозитория

Словари

While со списками и словарями

Кортежи

Работа с функциями

Парадигма функционального программирования

Чистые функции

Неизменяемые даные

Работа с простейшими алогоримами, к примеру, алгоритмом сортировки пузырьком