

O Object O Oriented P Programming

S1 = Python with Bianca

S2 = Java with Bryan

DT228(TU856)/DT282(TU858) - 2



**COMPUTER
SCIENCE**




Objectives

- Discuss the 3 worst lab test quiz questions
- Follow along with a possible lab test solution

git

When taking the question, it would appear as:


What is the connection between **git** and **gitHub**?

-  ☐ **gitHub** is a cloud service that manages **git** repositories.
-  ☐ **git** is a version control system.
-  ☐ If you want to work locally on your computer with a **gitHub** repository, you are required to have **git** installed on your computer.
- ☐ **gitHub** is a programme that you install locally on your computer.

git commands

When taking the question, it would appear as:

In the context of **gitHub/git** what do **git fork** and **git fetch** have in common?

- ☐ **git fork** and **git fetch** are both options to pull an update from an online repository on gitHub. **git fork** merges the change automatically with your local repository's master, while **git fetch** does not.
- ☐ **git fork** and **git fetch** are both options to upload your local changes to your own online gitHub repository. **git fork** merges the change automatically with your online repository's master, while **git fetch** does not.
-  ☒ These are two independent **git** commands that have nothing in common.
- ☐ **git fork** and **git fetch** both create a copy of someone else's online repository. However, only **git fork** can be used from within the gitHub platform's browser interface.

What's allowed as a dictionary key?

When taking the question, it would appear as:

Which of the following options correctly set up the dictionary variable **my_numbers**?



☒ my_numbers={"a":1, "b":2, "c":3}

☐ my_numbers={1:"a", 2:"b", 3:"c"}

☐ my_numbers=(1:"a", 2:"b", 3:"c")

☐ my_numbers=("a":1, "b":2, "c":3)

Coding Exercise

You are given the following two sentences:

You used to love cake but now you don't.

We used to enjoy cake but now we don't.

Write a Python **class** using **OOP principles** that prints a list of words that appear **exactly once** in one of the sentences, but not in the other sentence. Your solution should **not** be case sensitive.

Your output should be: **['love', 'enjoy']**

You can use

import collections

in order to use **collections.Counter** to count the words in the sentences. You do not manually have to populate a dictionary. **collections.Counter** takes a **list** as an argument and returns a **dictionary**.

Hint: use a **set** data type to compare the sentences for words that are not common to both sentences.