Lab 4: calculations, conditions and if

Refresher quiz 📝

Match the relational operator to its description

less than

equal to

greater than

not equal to

| == | > | != | < |

✓ Check   ❯

◎ ○ ○

I am confused ⚑

This week we began talking about conditional statements, Boolean expressions and `if` statements. Remember the syntax for an if statement is:
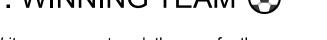
```
if condition:
    statements
elif condition:
    statements
else:
    statements
```

We also discussed relational operators. these are operators which compare 2 values and result in either True or False:

```
> < >= <= == (equal to) != (not equal to)
```

We will use these concepts in lab this week.

1. WINNING TEAM ⚽

Write a program to ask the user for the name and goals scored by 2 teams in a match. The program should then display the name of the winning team or tell the user if it was a draw. A

sample run might be:

```
What is the name of the first team ==> MTU
How many goals did MTU score ==> 3
What is the name of the second team ==> TUD
How many goals did TUD score ==> 2
MTU won the match.
```

Note: the team names are used in the prompts when we ask the user for the scores.

2. NEWSPAPER SUBSCRIPTION 📰

Write code to

1. Ask the user for the title of a newspaper
2. Ask the user for the monthly cost of the newspaper
3. Calculate the cost of an annual subscription to this newspaper
4. Ask the user if they have a student card. - If they do apply a 15% student discount
5. Display the details in a suitable format

3. NUMBER CHECKER 🧮

Write a program to ask the user for an integer number. The program should then check the following details about the number:

- Tell the user if the number is divisible by 7 (the modulus operator might help you here)
- Tell the user if the number is odd
  - If it is odd display its square root (math.sqrt() might help you here)
- Tell the user if the number is even
  - Display the square of the number

Remember that a number cannot be both odd and even, so if one is true we don't have to check the other.

4. COMMUTING TIME 🚗

This exercise might make use of // and % i.e. integer division and modulus operators

Ask the user for their name and the time in minutes they used to spend commuting to and from work each day.

The program should tell them how much time in days, minutes and hours they spent commuting per year.

Assume they work 5 days a week for 46 working weeks per year.

Show the answer in minutes in your first version. Later convert it to days, hours and minutes and display this as well.

If the user spends 5 or more days commuting per year show them a message to say that they have the option of working from home.

If the user spends less than 5 days commuting per year then show them a message congratulating them on reducing their carbon footprint.

```
Enter your name: Fred
Enter the time in minutes that you travel daily: 25
Fred commutes for 3 days, 23 hours and 50 minutes each year. Cong
ratulations! You have a low carbon footprint.
```

5. RANDOM GUESSING GAME 🔮

```python
import random
MIN = 1
MAX = 10
number = random.randint(MIN, MAX)
print(number)
```

1. The above code generates a random number between 1 and 10, copy and paste it into your IDE and run it to find out. Edit and add to this code so that you ask the user to guess the random number once and tell them if they guessed correctly or incorrectly.
2. Augment your program so that it now tells them if the number of greater than or less than the correct value if they guess the wrong number.

6. FLOATING POINT NUMBER CHECKER 👻

Read a floating-point number from the user and determine if it has a decimal part i.e. 5.45 has a decimal part (0.45) whereas 5.00 does not.

Print an appropriate message in each case.

Hint: is there a transformation we can do to the number which will give us information about the decimal part of the number? Consider what the int() and float() functions would do to the number.

7. GROUP EXERCISE 🤸 (please wait for me to initiate this exercise)

In exercise 1, you wrote code to determine the outcome of a match. If you were a lecturer and you were in charge of evaluating student's code how would you assign marks for this exercise? Consider the importance of code functionality (does the code do what it is meant to do?), code readability (choice of variable identifiers, comments, clear layout).

You have 10 marks to give for successful completion of this exercise, create a rubric for these 10 marks. Write your completed rubric down and clearly define each gradable component and the marks assigned to it.

Once you have completed your rubric discuss it with your neighbour. After your discussion use the rubric to evaluate your neighbour's code. Ensure you point out to good things that they did that gained them marks. Make a note of the improvements they could make in future iterations of their code.