



COMPUTING PROGRAMMING

PN - STUDENTS EVALUATION

Exam related course	Computing programming - Python
Exam Duration	2 h 00
Issue data	FEV 2023
Rules	<ul style="list-style-type: none">• Internet connection is forbidden• Students shall turn off internet during the exam• Chatting or talking to other students is forbidden• Only the list of allowed Python instruction is allowed for this exam (ALLOWED INSTRUCTIONS.pdf)

EXERCICES	POINT
Exercise 1	10
Exercise 2	30
Exercise 3	30
Exercise 4	30
TOTAL	100



EXERCISE 01 (10 pts)

Objective:

We want to know if a number is in the range [10, 20]

- Get a number from input
- Print "inside" if this number is in the range [10-20], or "outside" otherwise

Input – Outputs

INPUTS	Integer – <i>the number</i>
OUTPUT	String – – The result

Examples

INPUT	OUTPUT
14	inside
10	inside
20	inside
9	outside
-15	outside

Help?

See the **start code** (startCode.py)



EXERCISE 02 (30 pts)

Objective:

We want to know among two persons **which one is the youngest**

- The first person enters a year/month/day of birth
- The second person enters a year/month/day of birth
- Print out **whether the first of the second person is the youngest.**

Input – Outputs

INPUTS	Integer – <i>person 1 year of birth</i> Integer – <i>person 1 month of birth</i> Integer – <i>person 1 day of birth</i> Integer – <i>person 2 year of birth</i> Integer – <i>person 2 month of birth</i> Integer – <i>person 2 day of birth</i>
OUTPUT	String – print out whether the first of the second person is the youngest.

Examples

INPUT	OUTPUT
2002 12 31 2002 12 25	The first person is the youngest Explanations: -the first person is born on dec 31th 2002 -the second person is born on dec 25th 2002
2002 12 31 2002 12 31	Both persons have the same age! Explanations: -the first person is born on dec 31th 2002 -the second person is born on dec 31th 2002
2001 12 31 2002 12 31	The second person is the youngest Explanations: -the first person is born on dec 31th 2001 -the second person is born on dec 31th 2002

Help?

See the **start code** (startCode.py)



EXERCISE 03 (30 pts)

Objective:

We want to **filter a list of words** according to a specific rule.

- As an input, you have a list of names, for example:

["AAB", "CD", "AB", "BCCAA"]

- You need to keep only the names containing **2 A and 1 B**: for example, with above input, you should output only 2 words:

["AAB", "BCCAA"]

Input – Outputs

INPUT	Array of string
OUTPUT	Array of string

Examples

INPUT	OUTPUT
["AAB", "CD", "AB", "BCCAA"]	["AAB", "BCCAA"]
["A", "B", "C"]	[]
["A", "BBAAX", "BRONANA"]	["BAAX", "BRONANA"]

Help?

See the **start code** (startCode.py)



EXERCISE 04 (30 pts)

Objective:

We want to know the **first name** of the teachers who teaches **java** in **Philippines**

As an input, you have 2 arrays of dictionaries:

- The first one represents **the topics** for a given group, teacher, and country:

```
[
  {"topic": "java", "teacher-id": 32, "country": "Philippines"},
  {"topic": "java", "teacher-id": 33, "country": "Philippines"},
  {"topic": "html", "teacher-id": 30, "country": "Cambodia"},
  {"topic": "java", "teacher-id": 31, "country": "Cambodia"},
]
```

- The second one represents **the teacher's** information:

```
[
  {"teacher-id": "30", "first-name": "Him", "last-name": "Hey"},
  {"teacher-id": "31", "first-name": "Ronan", "last-name": "Ogor"},
  {"teacher-id": "32", "first-name": "Gran", "last-name": "Sabandal"},
  {"teacher-id": "33", "first-name": "Christian", "last-name": "Mediola"},
]
```

- From this example, 2 teacher are teaching in Philippines the topic Java : so you need to print:

```
[Gran, Christian]
```

Help?

See the **start code** (startCode.py)