

Programming Assignment 03

-

While loops

Instructions

This programming assignment consists of **3 programming exercises**.

You have to:

1. **download** the empty Python files on NYU Classes
2. **edit** them according to the assignment
3. **verify** on your computer that it works
4. **upload** them back on NYU Classes (**do not change the filenames**)

Exercise 1 - Checking user input

Write a program (in the file `exercise1.py`) that does the following:

- **continuously** asks the user to **input a score** which is an integer between 0 and 100 (inclusive)...
- ...**until** the user enters a correct number (**display a message** if the user did not enter a correct number. see sample examples)
- then finally **prints the corresponding grade**

Table of correspondences score/grade:

Grade	Minimum score (inclusive)
A+	96
A	92
A-	88
B+	84
B	81
B-	78
C+	75
C	72
C-	69
D+	66
D	63
F	0

Sample example 1 (the user input is in **red**, the printed output is in **blue**):

```
Score: 135
Invalid input. Please enter a score between 0 and 100.
Score: -25
Invalid input. Please enter a score between 0 and 100.
Score: 100
Your grade is A+
```

Sample example 2:

```
Score: 87
Your grade is B+
```

Sample example 3:

```
Score: 101
Invalid input. Please enter a score between 0 and 100.
Score: 71
Your grade is C-
```

Exercise 2 - Password generator (updated version)

Write a program (in the file `exercise2.py`) that does the following:

1. asks the user to **input** a **password length** `N` (type: `int`, positive)
2. then, generates and **prints** a random password of `N` characters.

The **valid possible characters** in the password are:

- lowercase letters `a` to `z`
- uppercase letters `A` to `Z`
- digits `0` to `9`
- and some special characters: `~`, ```, `!`, `@`, `#`, `$`, `%`, `^`, `&`, `*`, `(`, `)`, `_`, `-`, `+`, `=`, `{`, `[`, `}`, `]`, `|`, `\`, `:`, `;`, `"`, `'`, `<`, `,`, `>`, `.`, `?`, and `/`

Note

You **must** use a `while` loop for this exercise.

Sample examples (the user input is in **red**, the printed output is in **blue**):

Password length: **10**
i7W\$fk7X+x

Password length: **6**
>v\zoP

Exercise 3 - Password checker

In order to be valid your password must have at least one of each: lower case letter, upper case letter, digit and special character. **Write a program** (in the file `exercise3.py`) that does the following:

1. asks the user to **input a password**
2. then, prints: "This password is valid" if the password fulfills the conditions and prints: "This password is not valid" otherwise.

Note

You don't have to check all the characters in the password. Your program should stop as soon as the password is found valid.

Sample examples (the user input is in red, the printed output is in blue):

Enter a password: `i7W$fk7X+x`
This password is valid

Enter a password: `>v\zoP`
This password is not valid