

Python Programming

Lesson 6 – Input Evaluation and File Zipping

Lesson 6 - Outline

- Input Evaluation / Validation
- Zip and Unzip with Python

Input Evaluation / Validation

What is Input Evaluation

- **Background**

Collection of users' input, for example, for data analysis, decision making, etc.

- **Problem**

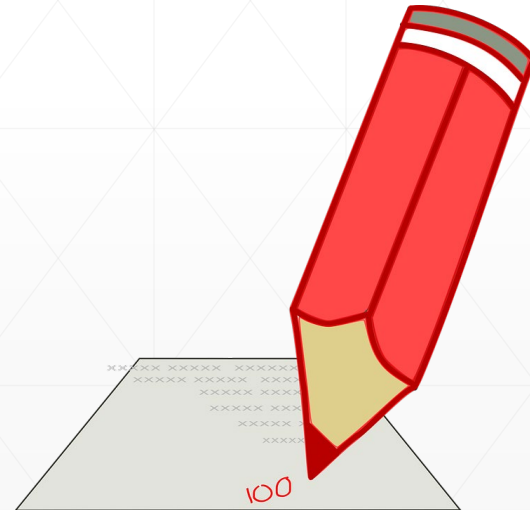
May collect bad quality of data!

- **Cause**

- Human error
- Wrong format of input

What is Input Evaluation

- **Let's try a standard input example (without Input Evaluation)**



What is Input Evaluation

- **Solution**

Evaluation / Validation for users input!



Common Types of Input Evaluation

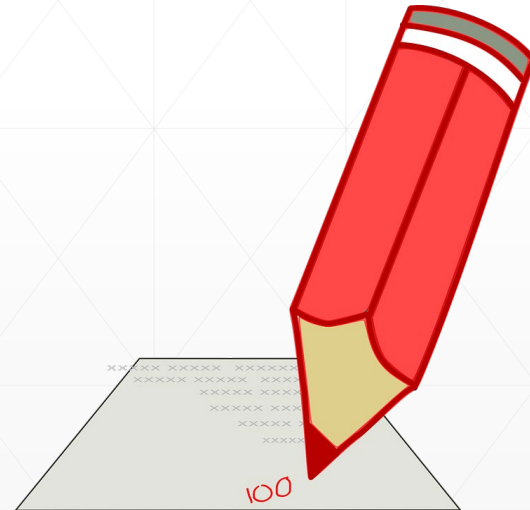
- Validation of Number format
- Validation of Text format
- Validation of Email Address
- Validation of specific format

Validation of Number format

Validation of Number format

1. How to make sure the input is a number?

Have a try first!



Validation of Number format

1. How to make sure the input is a number?

- Example

Use `int()` to validate whether the input can be converted to integer

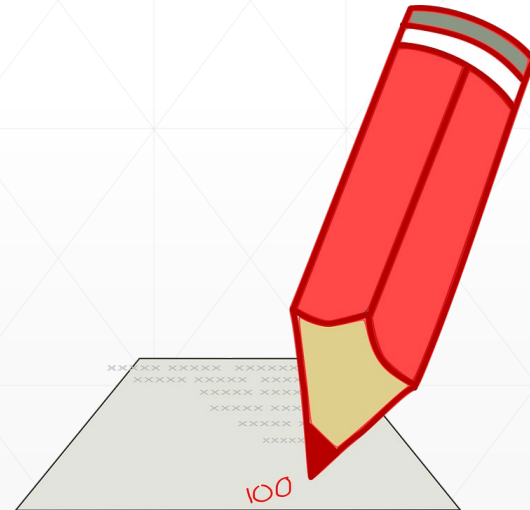
```
phone = int(input("Enter your phone number: "))  
print("Your Phone Number is " + str(phone))
```

Validation of Number format

1. How to make sure the input is a number?
2. Keep asking for user input if invalid

Have a try first!

Hint: Looping



Validation of Number format

1. How to make sure the input is a number?
2. Keep asking for user input if invalid

- Example

Use a while loop with the continue/break conditions, remember to use try-except to handle the exception!

```
while True:
    try:
        phone = int(input("Enter your phone number: "))
    except ValueError:
        print("Invalid input, enter a valid value without space")
        continue
    else:
        break

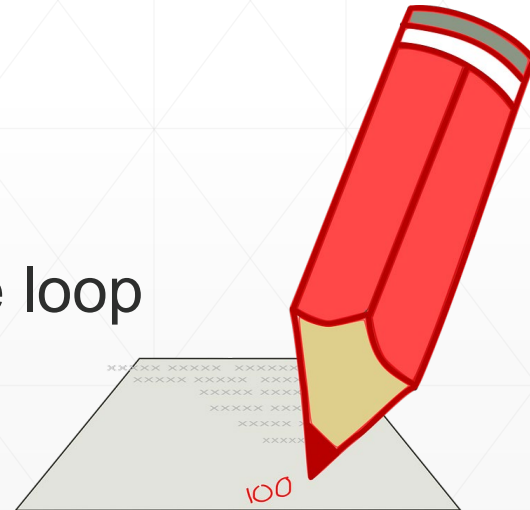
print("Your Phone Number is " + str(phone))
```

Validation of Number format

1. How to make sure the number is positive?
2. Keep asking for user input if invalid

Have a try first!

Hint: Use if condition within the loop



Validation of Number format

1. How to make sure the number is positive?
2. Keep asking for user input if invalid

- **Example**

Use an if condition to make sure the input number is a positive number!

```
while True:
    try:
        age = int(input("Please enter your age: "))
    except ValueError:
        print("Invalid input, enter a valid value")
        continue

    if age < 0:
        print("Your age must not be negative.")
        continue
    else:
        break

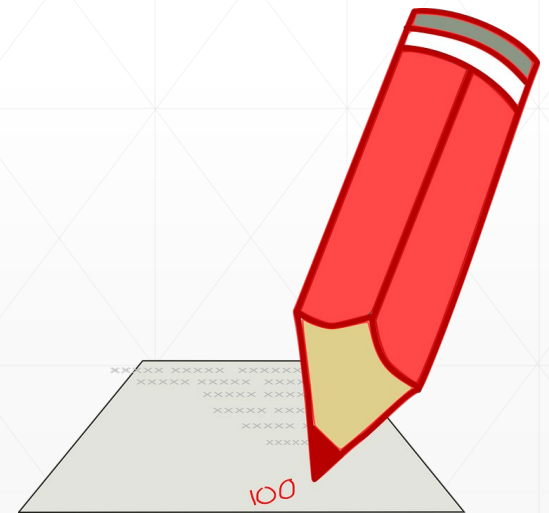
print("Your Age is " + str(age))
```

Validation of Number format

1. If age is equal or greater than 18, print “Here is your beer!”
2. Else print “Beer is not for you!”

Have a try first!

Hint: Where should the if-else locate?



Validation of Number format

1. If age is equal or greater than 18, print “Here is your beer!”
2. Else print “Beer is not for you!”

- **Example**

Use an if-else condition to print different wordings accordingly

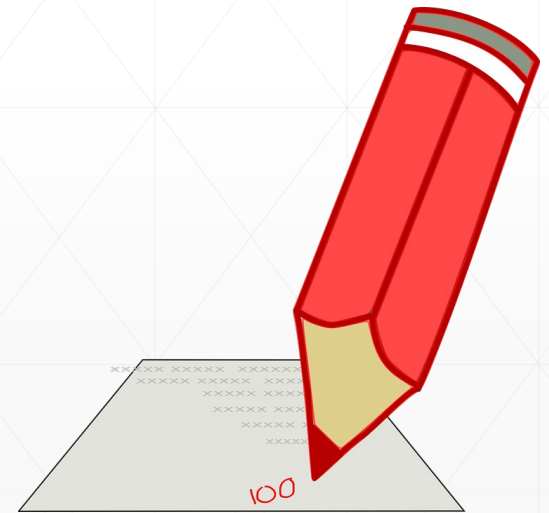
```
while True:
    try:
        age = int(input("Please enter your age: "))
    except ValueError:
        print("Invalid input, enter a valid value")
        continue
    if age < 0:
        print("Your age must not be negative.")
        continue
    else:
        break
if age >= 18:
    print("Here is your beer!")
else:
    print("Beer is not for you!")
```


Validation of Number format

1. How to make sure the length of input is certain number?

Have a try first!

Hint: Any functions can help?



Validation of Number format

1. How to make sure the length of input is certain number?

- Example

If the input is a string, directly use `len()` to check the length of the input. If not, use `str()` to convert the data type beforehand.

```
while True:
    try:
        phone = int(input("Please enter your phone number: "))
    except ValueError:
        print("Invalid input, enter a valid value")
        continue
    if len(str(phone)) != 8:
        print("Your phone number must be 8-digit.")
        continue
    else:
        break

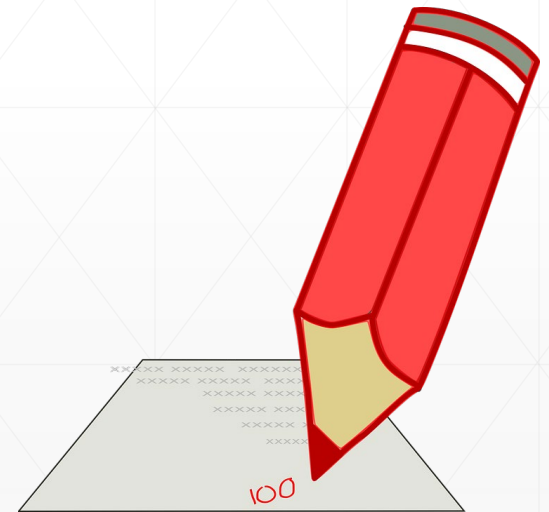
print("Your Phone Number is " + str(phone))
```

Validation of Text format

Validation of Text format

1. How to make sure the input text is certain value (e.g. M or F)?

Have a try first!



Validation of Text format

1. How to make sure the input text is certain value (e.g. M or F)?

- Example

Use if condition to check input matches the certain value or not

```
while True:
    gender = input("Please enter your gender (M or F): ")
    if gender != 'M' and gender != 'F':
        print("Invalid input")
        continue
    else:
        break
print("Your Gender is " + gender)
```

Validation of Email Address

Validation of Email format

1. How to validate a format of email...?

Let's learn about regular expression first!

What is Regular Expression

- **Definition**

Try to match the pattern with the provided string

- **Syntax**

```
re.match(pattern, string, flags=0)
```

- **Explanation**

| | |
|----------------|-------------------------------------------------------------------------------------------|
| pattern | The regular expression to be matched |
| string | The string which would be searched to match the pattern at the beginning of string |
| flags | Use of bitwise (<i>Less important in this course</i>) |



What is Regular Expression

- **More information**

- `[0-9]` => Match any digit; same as `[0123456789]`
- `[a-z]` => Match any lowercase letter
- `[A-Z]` => Match any uppercase letter
- `[a-zA-Z0-9]` => Match any of the pattern
- `[A-Z|a-z]{2,}` => A-Z or a-z with 2 or more characters

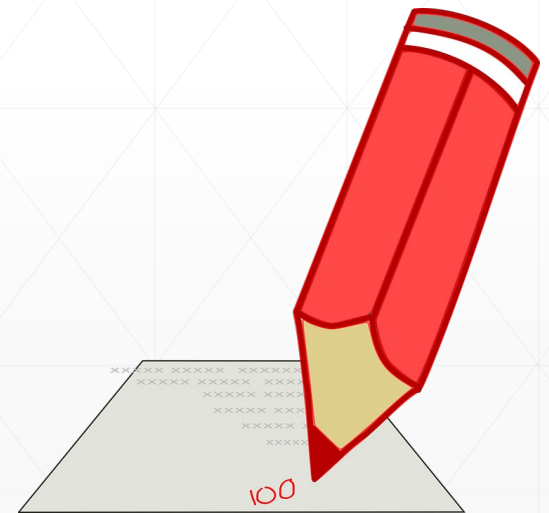
- **Example of email address validation**

`[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}`

Validation of Email format

1. How to validate a format of email...?

Let's try together!



Validation of Email format

1. How to validate a format of email...?

▪ Example

Use of regular expressions to define your pattern

```
# Import module for regular expressions
import re

regex = r'[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}'

email = input("Please enter your email: ")

if(re.fullmatch(regex, email)):
    print("Valid Email Format")
else:
    print("Invalid Email Format")
```

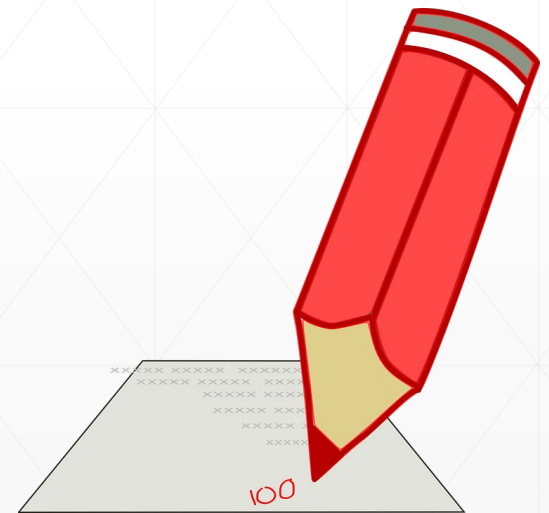
Validation of specific format

Validation of specific format

1. How to validate a format of HKID?

Challenge yourself!

Hint: Other than the pattern,
check digit is also a matter!



Validation using some modules

Validation using some modules

- We can perform the validation by some other modules ... e.g. `pyinputplus`, `validation` method in `django`

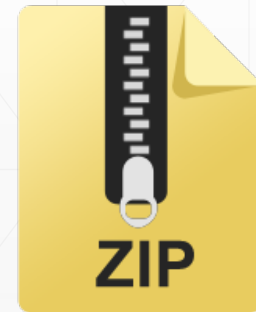
Search more if you have interest!

Zip and Unzip with Python

Zip and Unzip with Python

- **Background**

“Zip files are primarily used to reduce the size of a file by reducing the redundancy of information in that file”



Zip file/folder with Python

Zip with Python

- **Create the zip entire directory**

- `import shutil`
- `shutil.make_archive(output_filename, 'zip',
root_dir_name, base_dir_name)`

- **Explanation**

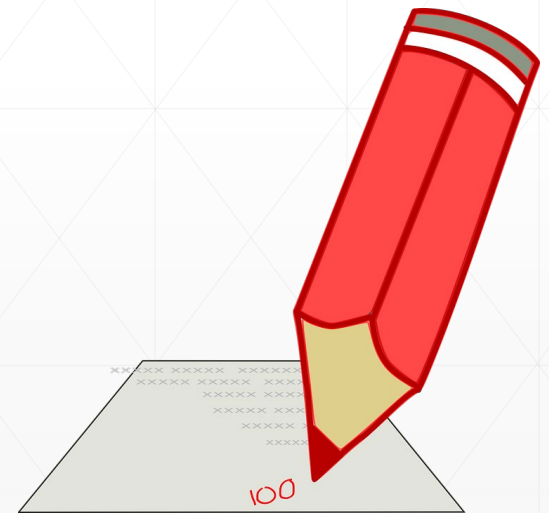
`output_filename`: the output zip file (with path)

`root_dir_name`: the directory that contains the files or
folders to be zipped

`base_dir_name`: the filename / folder name to be zipped

Zip with Python

- Let's have a try!



Unzip file/folder with Python

Unzip with Python

- **Unzip the zip file**

- `import shutil`
- `shutil.unpack_archive(zipfile_pathname, extract_destination)`

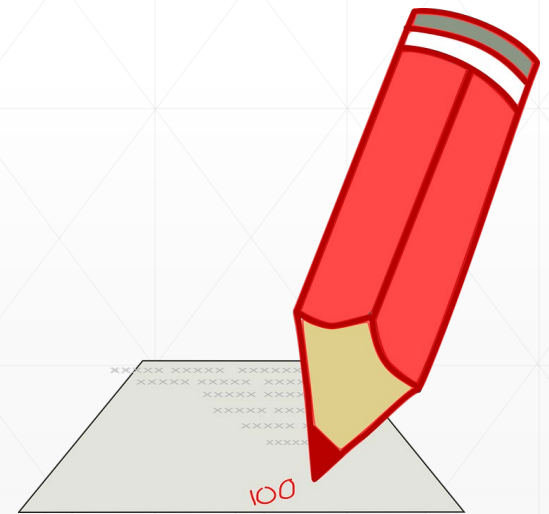
- **Explanation**

`zipfile_pathname`: the zip file (with path)

`extract_destination`: the destination directory to be unzipped

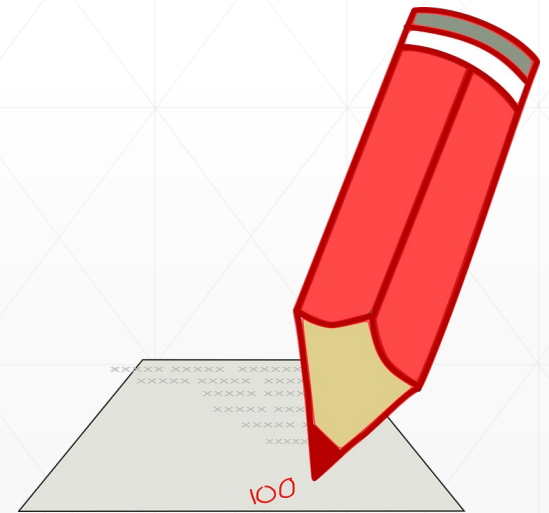
Unzip with Python

- Let's have a try!



Zip/ Unzip with Python

- Let's try some more examples!



Reference

Reference of make_archive

- https://docs.python.org/2/library/shutil.html#shutil.make_archive

Thank you