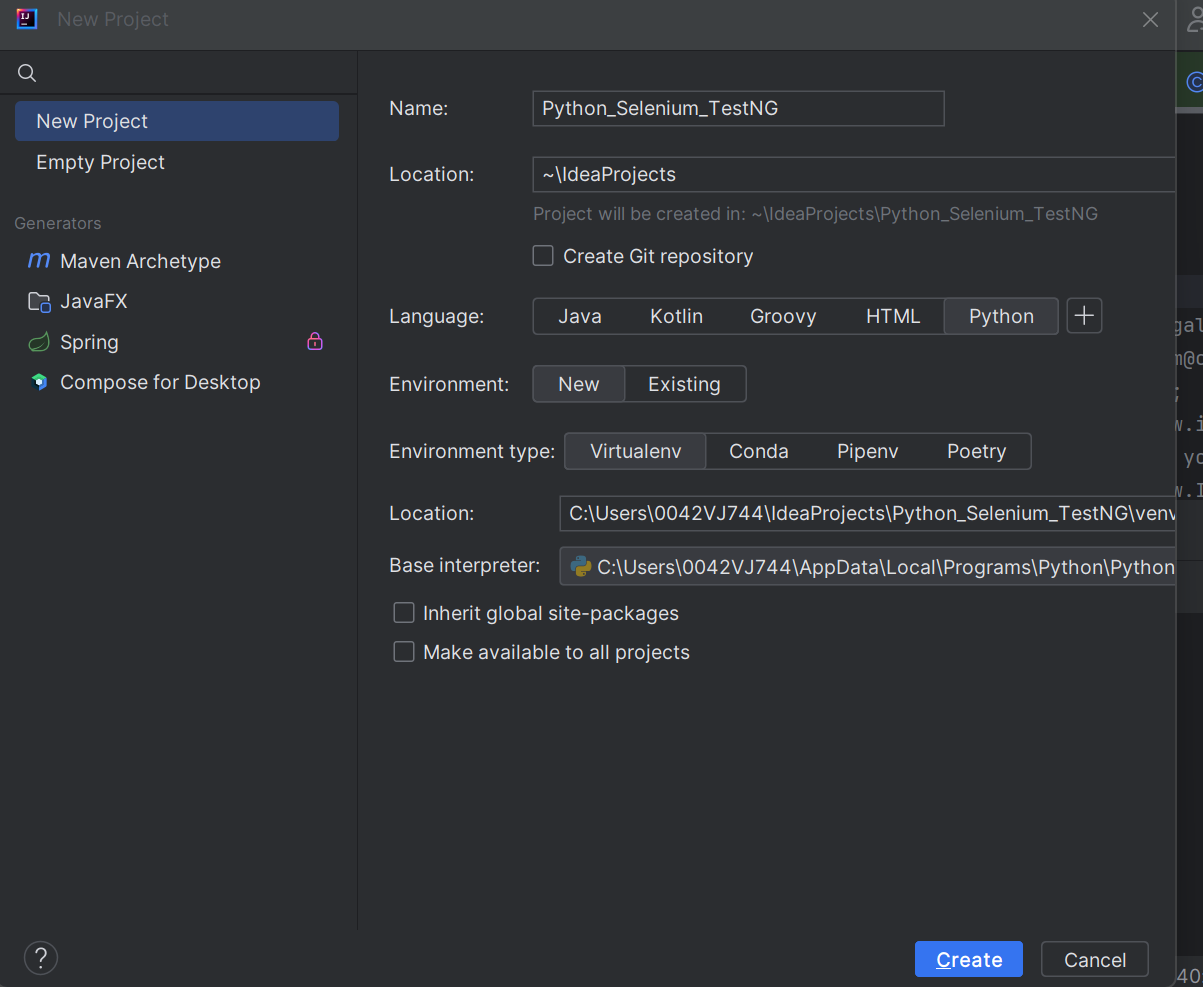
**Python session 1:**

*Install python plugin in intellij*

Install “Python community Edition” plugin

File >> setting >> plugin >> search for “Python community Edition”

**Setting:**



Python is different from java

You can reassign the values

Variable data type is assigned automatically based on input

#- single line comment

“”” or ‘’’ - commenting multiple line comment

X = 1 // int

F=2.9 //Float

C=2j+I // complex

print("hello world")  
s=100  
greetings="hello"  
print(s,greetings)  
k=10.50  
print(k)  
print(type(k))  
print(type(greetings))

Output:

hello world

100 hello

10.5

<class 'float'>

<class 'str'>

Replace

A=”hello,world”

Print(a.replace(“h”,”J”)) #output jello,world

Check string

txt = "The rain in Spain stays mainly in the plain"

x = "ain" in txt

y = "ain" not in txt

print(x) # True

print(y) # False

#output True

**Combine string**

age = 36

txt = "My name is John, I am " + age // it will error out this not possible

print(txt)

Instead you can make it as

age = 36

txt = "My name is John, I am " + str(age) // output My name is John, I am 36

*Example 1*

age = 36

txt = "My name is John, and I am {}"

print(txt.format(age))

# Output: My name is John, and I am 36

*Example 2*

quantity = 3

itemno = 567

price = 49.95

myorder = "I want {} pieces of item {} for {} dollars."

print(myorder.format(quantity, itemno, price))

# Output: I want 3 pieces of item 567 for 49.95 dollars.

**Example 3 with index value:**

quantity = 3

itemno = 567

price = 49.95

myorder = "I want to pay {2} dollars for {0} pieces of item {1}." // using index value

print(myorder.format(quantity, itemno, price))

# Output: I want to pay 49.95 dollars for 3 pieces of item 567.

**To stop execution :**

Syntax:

should\_i\_exit = true

if should\_i\_exit == true:

**raise SystemExit**

**Loop**

*Break - break the loop*

Example:

a = 0

while a < 5:

print(a)

if a == 3:

break

a += 1

# Output:

# 0

# 1

# 2

# 3

*Continue - continue the loop*

Example

a = 0

while a < 5:

a += 1

if a == 3:

continue

print(a)

# Output:

# 1

# 2

# 4

# 5

**For loop:**

fruits = ["apple", "banana", "cherry"]

for fruit in fruits:

print(fruit)

# Output:

# apple

# banana

# cherry

for x in "banana":

print(x)

# Output:

# b

# a

# n

# a

# n

# a

With the break statement we can stop the loop before it has looped through all the items:

fruits = ["apple", "banana", "cherry"]

for fruit in fruits:

if fruit == "banana":

**break**

print(fruit)

# Output:

# apple

fruits = ["apple", "banana", "cherry"]

for fruit in fruits:

if fruit == "banana":

**continue**

print(fruit)

# Output:

# apple

# cherry

**Nested loop**

adjectives = ["red", "big", "tasty"]

fruits = ["apple", "banana", "cherry"]

for adjective in adjectives:

for fruit in fruits:

print(adjective, fruit)

# Output:

# red apple

# red banana

# red cherry

# big apple

# big banana

# big cherry

# tasty apple

# tasty banana

# tasty cherry

**Joining list/ concatenate**

List 1 =[ 1,2,3]

List 2=[4,5,6]

List 3= list 1+list 2

Copy list

Apend list

Insert list

Lst.insert(1,cherry) // in index 1 cherry will be inserted