1. Define three functions fun(), disp() and msg(). Store them in a list and call them one by one in a loop.

INPUT

```
def fun():
    print("Function fun() called")

def disp():
    print("Function disp() called")

def msg():
    print("Function msg() called")

functions = [fun, disp, msg]
for func in functions:
    func()
```

OUTPUT

```
Function fun() called
Function disp() called
Function msg() called
```

2. Suppose there are two lists, one containing numbers from 1 to 6, and other containing numbers from 6 to 1. Write a program to obtain a list that contains elements obtained by adding corresponding elements of the two lists. (hint: use map and lambda functions)

INPUT

```
list1 = [1, 2, 3, 4, 5, 6]
list2 = [6, 5, 4, 3, 2, 1]
result = list(map(lambda x, y: x + y, list1, list2))
print("Resultant List:", result)
```

OUTPUT

```
Resultant List: [7, 7, 7, 7, 7, 7]
```

3. Generate the list of 10 different random numbers between -15 and 15. Create a new list by obtaining square of all numbers in a list.

INPUT

```
import random

random_numbers = random.sample(range(-15, 16), 10)  # Generate 10 random numbers
squares = [num ** 2 for num in random_numbers]
print("Random numbers:", random_numbers)
print("Squares:", squares)
```

OUTPUT

```
Random numbers: [-11, -6, 0, 8, 5, -1, 7, -10, 3, 15]
Squares: [121, 36, 0, 64, 25, 1, 49, 100, 9, 225]
```

4. Consider the following list:
    lst = ['madam','Python',"malayalam",12321]
    Write a program to print those strings which are palindromes.

INPUT

```
lst = ['madam', 'Python', "malayalam", 12321]
palindromes = [item for item in lst if str(item) == str(item)[::-1]]
print("Palindromes:", palindromes)
```

OUTPUT

```
Palindromes: ['madam', 'malayalam', 12321]
```

5. A list contains names of Faculty Members. Write a program to filter out those names whose length is more than 8 characters.

INPUT

```
faculty_members = ["Alexander", "John", "Catherine", "Elizabeth", "Mike"]
filtered_names = [name for name in faculty_members if len(name) > 8]
print("Names with length > 8:", filtered_names)
```

OUTPUT

```
Names with length > 8: ['Alexander', 'Catherine', 'Elizabeth']
```