1.

```
Import random
num = random.randint(0, 1000)
mod = num % 2
if mod > 0:
   print("You picked an odd number.")
else:
   print("You picked an even number.")
```

2.

```
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
num = int(raw_input("Choose a number: "))
new_list = []

for i in a:
        if i < num:
                new_list.append(i)
print new_list

# or:
new_list = [ele for ele in a if ele < num]
print(new_list)
```

3.

```
from math import pi
r = float(input ("Input the radius of the circle : "))
print ("The area of the circle with radius " + str(r) + " is: " + str(pi * r**2))
```

4.

```
n=int(input("Input a number "))
d = dict()

for x in range(1,n+1):
   d[x]=x*x

print(d)
```

5.

```
a = [1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89]
b = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13]

# use for loop
new_list = []
for element in a:
    if element in b and element not in new_list:
        new_list.append(element)

# use python set
new_list = list(set(a).intersection(set(b)))

# list comprehension
new_list = [ele for ele in set(a) if ele in set(b)]

# generate random 10 element integer array in range of 0-99
import random
rand_array = random.sample(range(99), 10)
```

6.

```
possible_vowel = ['a', 'e', 'i', 'o', 'u']
Str_list = []

# using for loops
for vowel0 in possible_vowel:
    for vowel1 in possible_vowel:
        if vowel1 != vowel0:
            for vowel2 in possible_vowel:
                if vowel2 != vowel1 and vowel2!=vowel0:
                    Str_list.append(vowel0 + vowel1 + vowel2)

# or, use list comprehension:
Str_list = [v0 + v1 + v2 for v0 in possible_vowel for v1 in possible_vowel for v2 in
possible_vowel if v1!=v0 if v2!=v1 if v2!=v0]

# you can also using python module itertools, see itertools.combinations
```

7.

8.

```
def gen_strobogrammatic(n):
    """
    :type n: int
    :rtype: List[str]
    """
    result = helper(n, n)
    return result

def helper(n, length):
    if n == 0:
        return [""]
    if n == 1:
        return ["1", "0", "8"]
    middles = helper(n-2, length)    # recursion
    result = []
    for middle in middles:
        if n != length:
            result.append("0" + middle + "0")
        result.append("8" + middle + "8")
        result.append("1" + middle + "1")
        result.append("9" + middle + "6")
        result.append("6" + middle + "9")
    return result

print("n = 2: \n",gen_strobogrammatic(2))
print("n = 3: \n",gen_strobogrammatic(3))
print("n = 4: \n",gen_strobogrammatic(4))
```