

# SESSDSA16课程第一次上机作业

## ( Python入门 )

说明：本次作业分为两部分。为方便批改作业，请同学们在完成作业时遵守下面规则：

- (1) 请将第一部分作业的 Python 代码全部放入同一个文件，命名为 `dsa1_1.py`，同样，第二部分代码放入 `dsa1_2.py` 文件中，上交作业时将两个文件一起打包，命名为 `dsa1_学号_姓名.7z/.zip/.rar`；
- (2) 如果作业中对相关类或函数有明确命名/参数/返回值要求的，请严格按照要求执行；
- (3) 有些习题会对代码的编写进行特殊限制，请注意这些限制并遵守；
- (4) 作业在 3 月 17 日 24 时前提交到 `sessdsa@163.com` 信箱。

## 1. 基础部分

1. 创建一个函数，接受一个字符串和一个正整数 `n` 作为参数，返回把原字符串字符位置向右移动 `n` 个字符的字符串，例如接受的参数是 "abcd" 和 1，返回的字符串是 "dabc"；接受的参数是 "mnbol" 和 2，返回的字符串是 "olmnb"。

```
def dsa1_reverse(s,n)
```

2. 创建一个函数，接受一个正整数 `n` 作为参数，返回  $s=1!+2!+3!+\dots+n!$  的值。

```
def dsa1_factorialSum(n)
```

3. 创建一个函数，接受两个长度相同的元组，用这两个元组中的所有数据组成一个字典并返回。如元组 (1, 2, 3) 和 ("abc", "def", "ghi") 生成字典 {1:"abc", 2:"def", 3:"ghi"}。

```
def dsa1_generateDict(keys, values)
```

4. 创建一个函数，接受一个英文字符串作为参数，返回该字符串的整数表示。如输入 "eight-nine", 返回 89, 输入 "one-two-three-four-five", 返回 12345。

```
def dsa1_en2num(s)
```

5. 创建一个函数，接受两个字符串作为参数，返回两个字符串字符集合的并集。如接受的两个字符串为 "abc" 和 "bcd", 返回 `set(['a', 'b', 'c', 'd'])`。

```
def dsa1_getUnion(s1,s2)
```

6. 创建一个函数，接受两个参数 `y` 和 `m`，分别表示年和月，返回此年此月的天数。（如大月有 31 天，小月有 30 天，而闰年的 2 月有 29 天，平年则只有 28 天，年份如果能被 4 整除但不能被 100 整除或者能被 400 整除为闰年）

```
def dsa1_getDays(y,m)
```

7. 创建一个函数，接受一个参数  $n(n \geq 100)$ ，判断这个数是否为水仙花数（即满足如果这个数为  $m$  位数，则每个位上的数字的  $m$  次幂之和等于它本身，例如  $1^3 + 5^3 + 3^3 = 153$ ， $1^4 + 6^4 + 3^4 + 4^4 = 1634$ ），返回 True 或者 False。

```
def dsa1_isNarcNum(n)
```

8. 创建一个函数，接受一个参数  $max(max \geq 1000)$ ，调用第 7 题编写的函数，求 100 到  $max$  之间的水仙花数，并且输出到一个名为 `narcissistic.txt` 的文件中，每个数作为一行输出。

```
def dsa1_writeNarcNum(max)
```

9. 创建一个函数，读入第 8 题中输出的文件 `narcissistic.txt`，以列表的形式返回文件中的水仙花数，列表的每个元素类型为 `int` 型，例如返回 `[153,370,371]`。

```
def dsa1_readNarcNum()
```

10. 创建一个函数，接受一个参数  $n$ ，在屏幕上打印出杨辉三角形的前  $n$  行，如下打印了 6 行：

```
1
1 1
1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
```

```
def dsa1_printTri(n)
```

## 2. OOP 部分

1. 创建一个 `People` 类，`People` 的属性有 `name` 和 `age`，两个参数在实例化的时候传给构造器，`People` 类的方法有 `getName` 和 `getAge`，分别返回姓名和年龄。

```
p= People(name,age)
```

```
def getName(self)
```

```
def getAge(self)
```

2. 创建一个 `Student` 类，`Student` 类继承上一题中的 `People` 类，并且增加了属性 `sno`，`sno` 表示学号，和增加了方法 `getSno`。

```
stu= Student(name,age,sno)
```

```
def getSno(self)
```

3. 创建一个 `Xdict` 类，继承自 `dict` 类(python 中的字典类)，使得 `Xdict` 类支持 `dict` 类的所有操作，并且增加一个方法 `getKeys`，用于返回给定值对应的键的列表。(例如 `Xdict` 一个实例为 `xd=Xdict({2:"a", 3:"a", 4:(2,3)})`，则 `xd.getKeys("a")`返回值为`[2,3]`)

```
class Xdict(dict):  
    def getKeys(self, value):  
        .....
```