

```
In [22]: class RestaurantCheck:
def __init__(self, check_number, sales_tax_percent, subtotal, table_number, server_name):
    self.check_number = check_number
    self.sales_tax_percent = sales_tax_percent
    self.subtotal = subtotal
    self.table_number = table_number
    self.server_name = server_name

def calculate_total(self):
    return (self.subtotal * (1 + self.sales_tax_percent / 100))

def print_check(self):
    output_file = open('check' + str(self.check_number) + '.txt' , 'w')
    print('Check Number : ' , self.check_number , file = output_file)
    print('Sales tax      : ' , self.sales_tax_percent, '%' , sep=' ' , file = output_file)
    print('Subtotal       : {:.2f}'.format(self.subtotal) , file = output_file)
    print('Total          : {:.2f}'.format(self.calculate_total()) , file = output_file)
    print('Table Number : ' , self.table_number , file = output_file)
    print('Server Name  : ' , self.server_name , file = output_file)
    output_file.close()

check = RestaurantCheck(443,6,23.14,17,"Sonic the Hedgehog")
check.print_check()
```

```
In [6]: import re

phone_no = ["412345678", "9123456781" , "9456346452"]
def valid(no):
    no_pattern = re.compile(r'^\d{10}$')
    return no_pattern.match(no)
for i in phone_no:
    if(valid(i)):
        print("Phone Number with 10 digits" , i)

name = ("sample", "sample1", "sample2", "Hasa")
def mat(nam):
    nam_pattern = '[A-Z]+[a-z]+$'
    return re.search(nam_pattern, nam)
for i in name:
    if (mat(i)):
        print("Firt letter upper case Names are" , i)

email = ("sample@uk", "sample@gmail.com", "sample.com", "sample2@ourex.org")
def check(em):
    email_pattern = r'\b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,7}\b'
    if (re.findall(email_pattern, em)):
        print(em, "Is a Valid Email")
for i in email:
    check(i)

dataex= ("2013-22-23", "22-03-2021", "2022/22/12", "22/11/2020")
def dcheck(dt):
    data_pattern = re.compile('^(([0]?[1-9]|[1]2|[0-9]|[3][0|1])[-.](([0]?[1-9]|[1][0-2])[-.](([0-9]{4}|[0-9](2))$))$')
    return data_pattern.match(dt)
for j in dataex:
    if (dcheck(j)):
        print("Date with DD-MM-YYYY format is" , j)
```

```
Phone Number with 10 digits 9123456781
Phone Number with 10 digits 9456346452
Firt letter upper case Names are Hasa
sample@gmail.com Is a Valid Email
sample2@ourex.org Is a Valid Email
Date with DD-MM-YYYY format is 22-03-2021
```