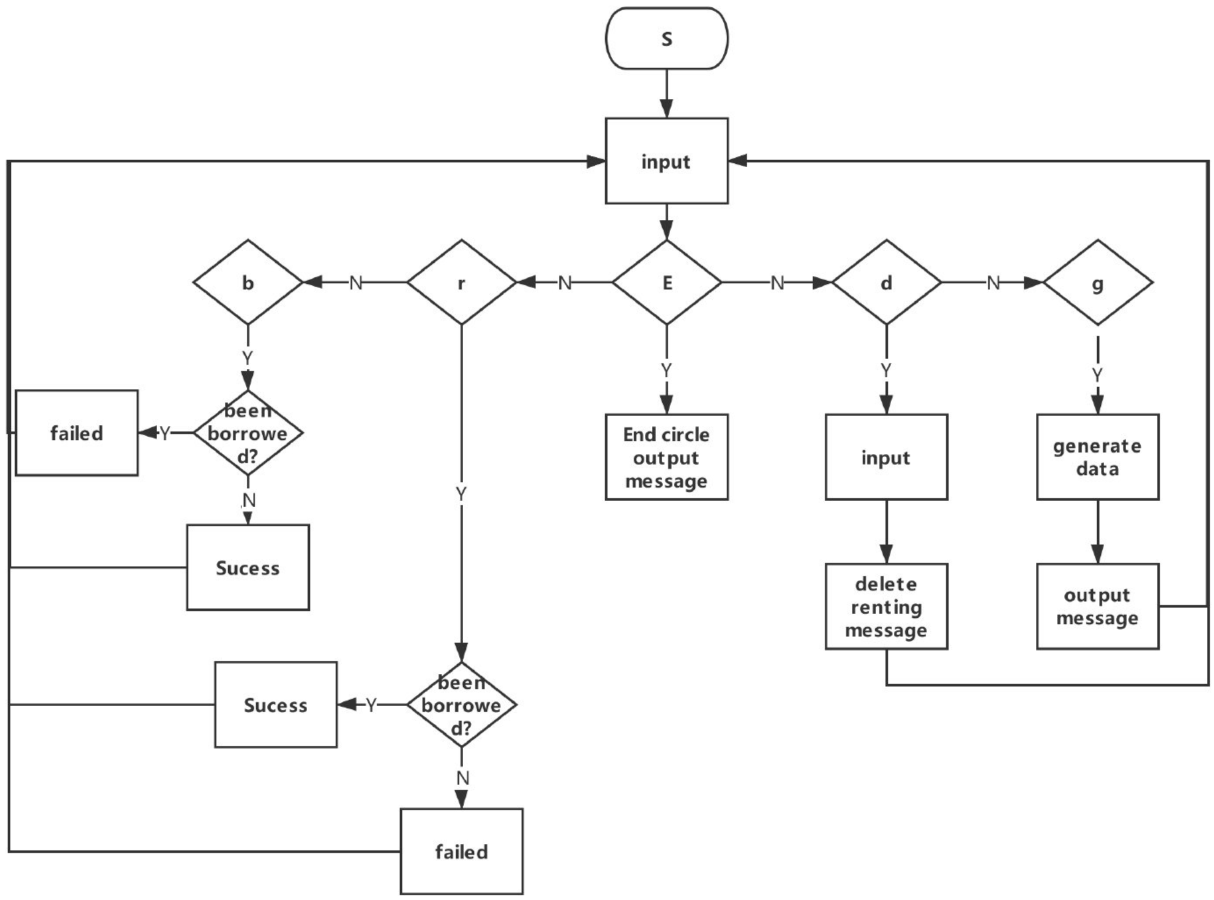
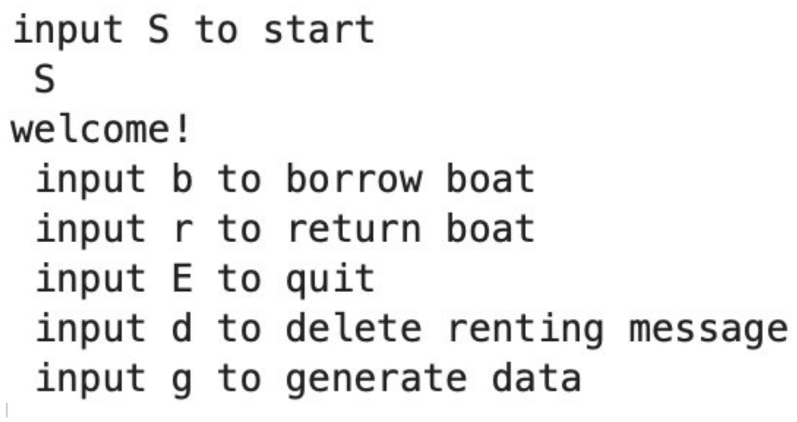
Flow chart



Demo interface

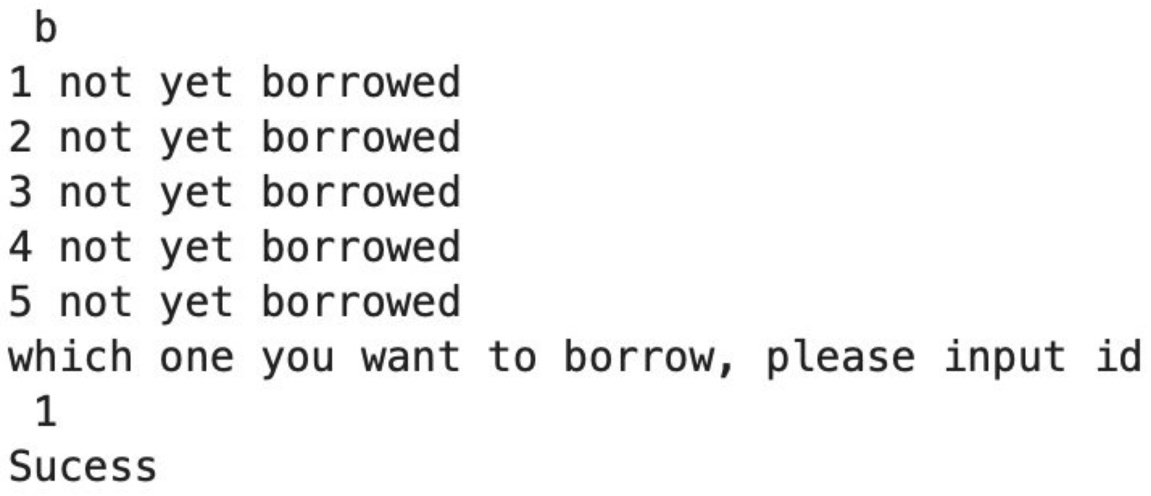
Welcome interface



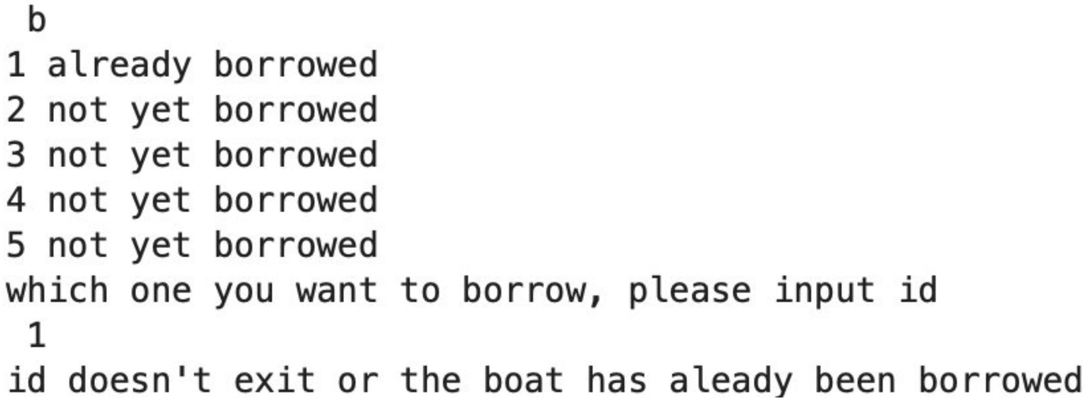
```
input S to start
 S
welcome!
 input b to borrow boat
 input r to return boat
 input E to quit
 input d to delete renting message
 input g to generate data
```

Borrow boat

The action will success only when the boat has not been borrowed



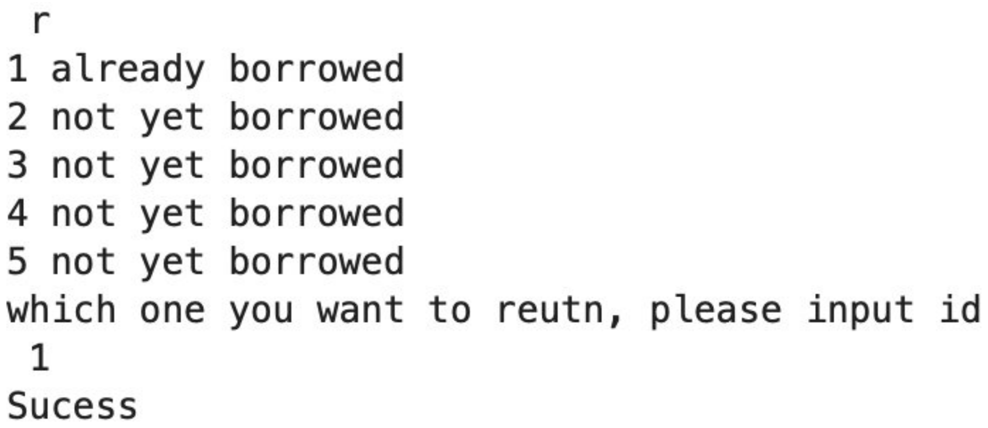
```
  b
 1 not yet borrowed
 2 not yet borrowed
 3 not yet borrowed
 4 not yet borrowed
 5 not yet borrowed
 which one you want to borrow, please input id
  1
 Sucess
```



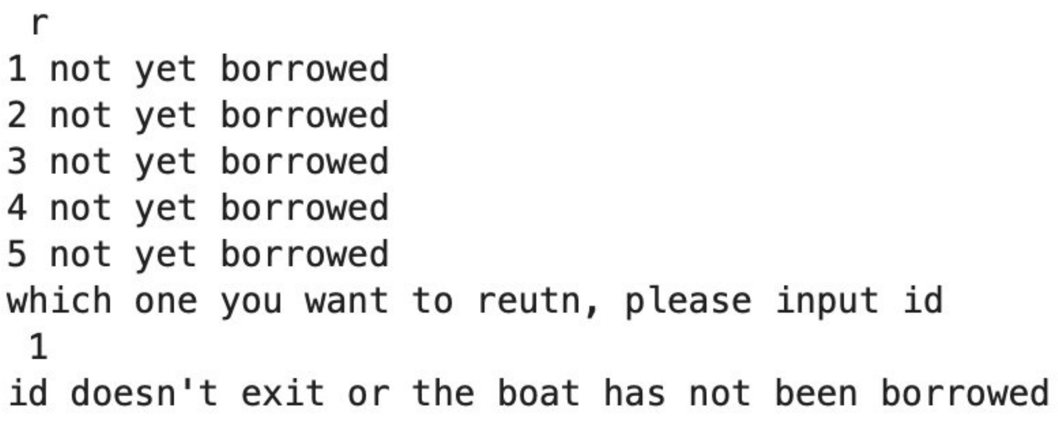
```
  b
 1 already borrowed
 2 not yet borrowed
 3 not yet borrowed
 4 not yet borrowed
 5 not yet borrowed
 which one you want to borrow, please input id
  1
 id doesn't exit or the boat has aleady been borrowed
```

return boat

The action will success only when the boat has been borrowed

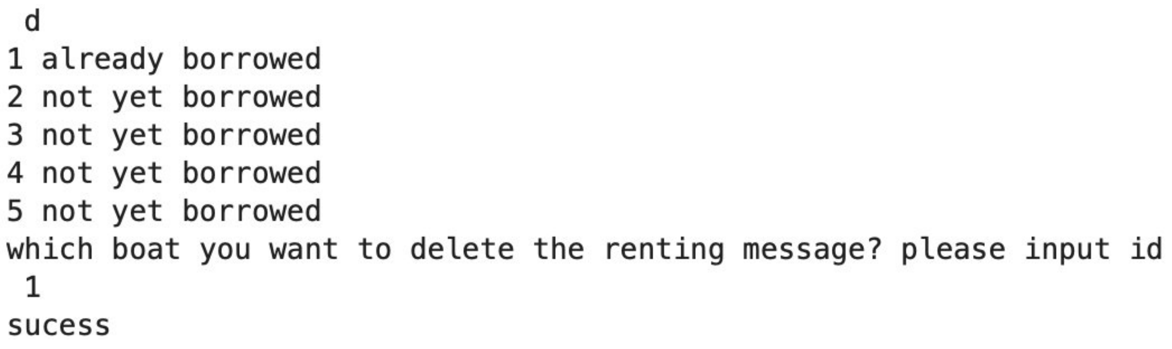


```
  r
 1 already borrowed
 2 not yet borrowed
 3 not yet borrowed
 4 not yet borrowed
 5 not yet borrowed
 which one you want to reutn, please input id
  1
 Sucess
```



```
 r
1 not yet borrowed
2 not yet borrowed
3 not yet borrowed
4 not yet borrowed
5 not yet borrowed
which one you want to reutn, please input id
 1
id doesn't exit or the boat has not been borrowed
```

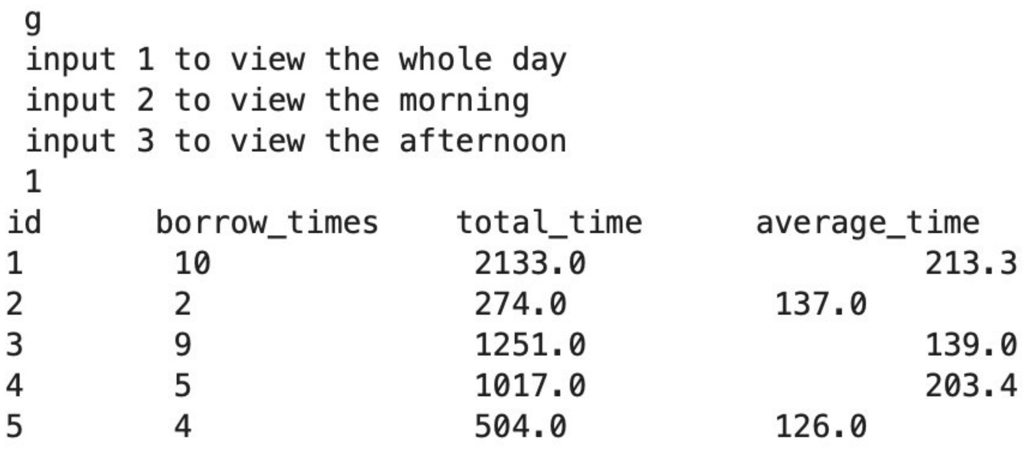
delete renting message



```
 d
1 already borrowed
2 not yet borrowed
3 not yet borrowed
4 not yet borrowed
5 not yet borrowed
which boat you want to delete the renting message? please input id
 1
sucess
```

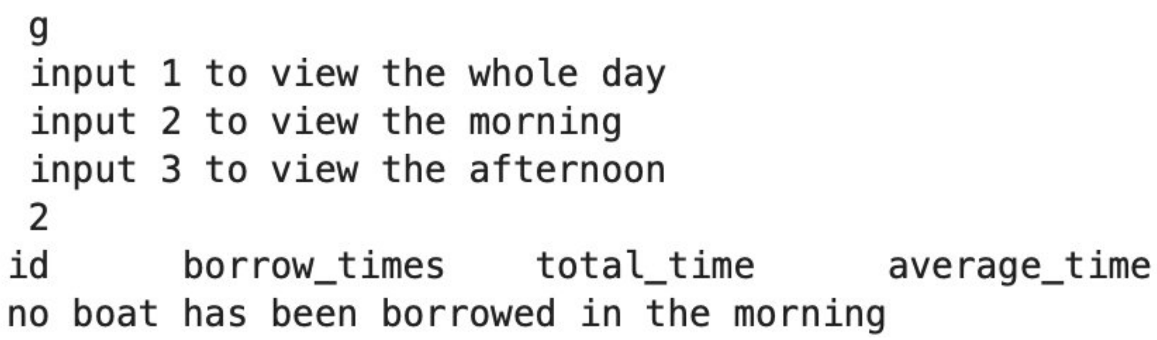
generate data

view the whole day



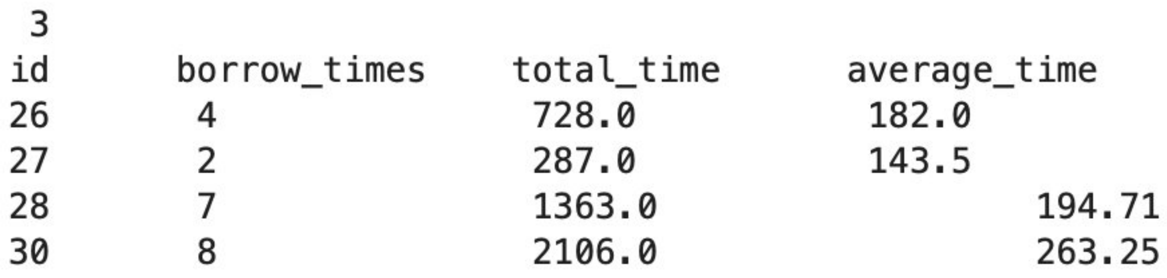
```
 g
 input 1 to view the whole day
 input 2 to view the morning
 input 3 to view the afternoon
 1
id       borrow_times     total_time       average_time
1         10               2133.0                   213.3
2         2                274.0           137.0
3         9                1251.0                   139.0
4         5                1017.0                   203.4
5         4                504.0           126.0
```

view the morning



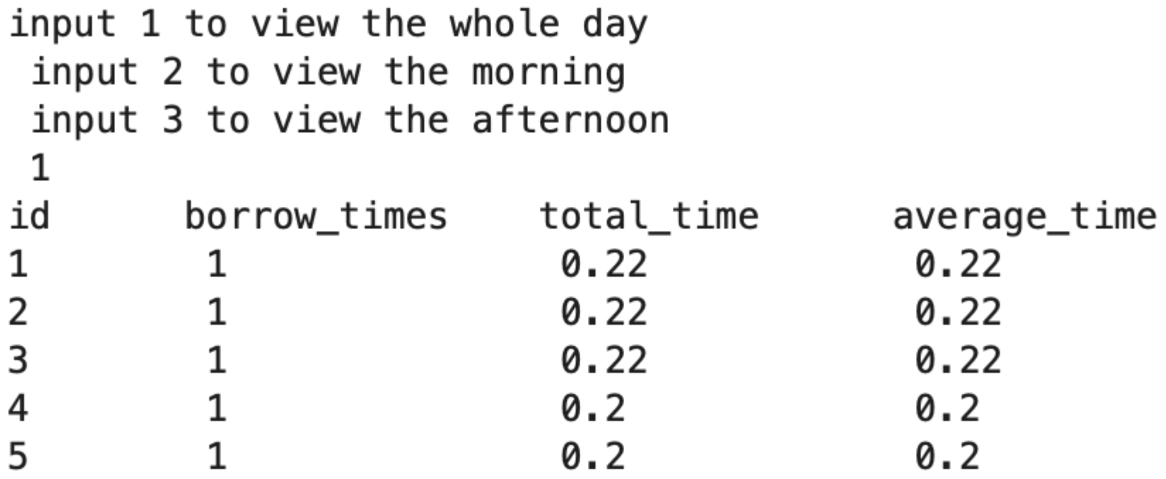
```
 g
 input 1 to view the whole day
 input 2 to view the morning
 input 3 to view the afternoon
 2
id      borrow_times    total_time      average_time
no boat has been borrowed in the morning
```

view the afternoon



```
  3
 id      borrow_times    total_time      average_time
 26       4               728.0           182.0
 27       2               287.0           143.5
 28       7               1363.0                  194.71
 30       8               2106.0                  263.25
```

End the circle



```
input 1 to view the whole day
 input 2 to view the morning
 input 3 to view the afternoon
 1
id      borrow_times    total_time      average_time
1        1               0.22            0.22
2        1               0.22            0.22
3        1               0.22            0.22
4        1               0.2             0.2
5        1               0.2             0.2
```

Code

```
from datetime import timedelta, datetime
import random
```

```
class boat:
    __id = 1
    def __init__(self, rent=5):
        # 船的编号
        self.id = boat.__id
        boat.__id += 1

        # 租金
        self.rent = rent

        # 记录这艘船当天的借出最长时间
        self.longest_time = 0
        self.longest_period = 0

        # 记录一天中总的借出时间
        self.total_time = 0


        # datetime
        self.start = 0

        # datetime
        self.end = 0

        # 记录是否被借出
        self.borrowed = False

        # 记录一天总的借出次数
        self.times = 0

        # 记录所有借出归还时间
        self.borrow_attr =[]

    def borrow_boat(self, start):
        # 新的一天 清零
        if self.start == 0 or start.day != self.start.day:
            self.longest_time = 0
            self.total_time = 0
            self.longest_period = 0
            self.times = 0

        if self.borrowed == False:
            self.borrowed = True
            self.start = start
            self.end = 0
        else:
            print('wrong,the boat has been borrewed')
    def return_boat(self, end):
        self.borrowed = False
        self.end = end
        self.times += 1
        period = (self.end - self.start).seconds / 60
        self.total_time += period
        if period > self.longest_time:
            self.longest_time = period
            self.longest_period = [self.start, self.end]

        attr = [self.start, self.end]
        self.borrow_attr.append(attr)


    def delete_renting_message(self):
        self.start = 0
        self.borrowed = False

    def print_longest_rentingtime(self,year=None, month=None, day=None):
        if year == None:
            print(self.longest_period[0],'-', self.longest_period[1],
self.longest_time, 'minute')
        else:
            longest_period = []
            longest_time = 0
            for periods in self.borrow_attr:
                if periods[0].year == year and periods[0].month == month and
periods[0].day == day:
                    temp = (periods[1] - periods[0]).seconds / 60
                    if temp > longest_time:
                        longest_time = temp
                        longest_period = [periods[0], periods[1]]
            print(longest_period[0],'-',longest_period[1],longest_time,
'minute')


    def print_message(self, year=None, month=None, day=None, all_day = 1):
        # all_day = 1 为全天 2为上午 3为下午
        if year == None:
            year=self.end.year
            month=self.end.month
            day=self.end.day

        for periods in self.borrow_attr:
            if periods[0].year == year and periods[0].month == month and
periods[0].day == day:
                if all_day == 1:
                    print(all_day)
                    print(periods[0],'-',periods[1])
                if all_day == 2:
                    if periods[0].hour < 12:
                        print(periods[0],'-',periods[1])
                if all_day == 3:
                    if periods[0].hour >= 12:
                        print(periods[0],'-',periods[1])
```

```
def quit(boats, all_day = 1):
    # all_day = 1 为全天 2为上午 3为下午
    print('id\tborrow_times\ttotal_time\taverage_time')
    sucess = False
    if all_day == 1:
        for boat in boats:
            if boat.times != 0:
                print(boat.id,'\t',boat.times,'\t\t',
round(boat.total_time,2),'\t\t', round(boat.total_time / boat.times,2))
            else:
                print(boat.id, 'has not been borrowed')
    elif all_day == 2:
        for boat in boats:
            if boat.times != 0 and boat.end.hour <12 :
                sucess = True
                print(boat.id,'\t',boat.times,'\t\t',
round(boat.total_time,2),'\t\t', round(boat.total_time / boat.times,2))
        if not sucess:
            print('no boat has been borrowed in the morning')

    elif all_day == 3:
        for boat in boats:
            if boat.times != 0 and boat.end.hour>=12:
                print(boat.id,'\t',boat.times,'\t\t',
round(boat.total_time,2),'\t\t', round(boat.total_time / boat.times,2))
                sucess = True
        if not sucess:
            print('no boat has been borrowed in the afternoon')

def deleted(boats):
    print('which boat you want to delete the renting message? please input
id')
    message = int(input())
    sucess = False
    for boat in boats:
        if boat.id == message and boat.borrowed == True:
            sucess = True
            boat.delete_renting_message()
            print('sucess')
    if not sucess:
        print("id doesn't exit or the boat has not been borrowed")

def generate_data(boats):
    for boat in boats:
        borrow_times = random.randint(1,10)
        for t in range(borrow_times):
            borrow_hour = random.randint(7,18)
            borrow_minute = random.randint(0,59)
            return_hour = random.randint(borrow_hour+1,19)
            return_minute = random.randint(0,59)
            start = datetime(2019,9,22,borrow_hour,borrow_minute)
            end = datetime(2019,9,22,return_hour, return_minute)
            boat.borrow_boat(start)
            boat.return_boat(end)

def welcome():
    boats = []
    for i in range(5):
        boats.append(boat())
    print('input S to start')
    message = input()
    if message == 'S':
        while(message != 'E'):
            print('welcome!\n', 'input b to borrow boat\n input r to return
boat\n input E to quit\n input d to delete renting message\n input g to
generate data')
            message = input()
            if message != 'E' and message != 'g':
                for bo in boats:
                    print(bo.id, end = ' ')
                    if bo.borrowed == True:
                        print('already borrowed')
                    else:
                        print('not yet borrowed')

                if message == 'b':
                    sucess = False
                    print('which one you want to borrow, please input id')
                    message = int(input())
                    for bo in boats:
                        if bo.id == message and bo.borrowed == False:
                            bo.borrow_boat(datetime.now())
                            sucess = True
                            print('Sucess')
                            break
                    if not sucess:
                        print("id doesn't exit or the boat has aleady been
borrowed")
```

```
                if message == 'r':
                    sucess = False
                    print('which one you want to reutn, please input id')
                    message = int(input())
                    for bo in boats:
                        if bo.id == message and bo.borrowed == True:
                            bo.return_boat(datetime.now())
                            print('Sucess')
                            sucess = True
                            break
                    if not sucess:
                        print("id doesn't exit or the boat has not been
borrowed")

                if message == 'd':
                    deleted(boats)

            if message == 'E':
                print(' input 1 to view the whole day\n input 2 to view the
morning\n input 3 to view the afternoon')
                message = int(input())
                quit(boats, message)
                message = 'E'

            if message == 'g':
                generate_data(boats)
                print(' input 1 to view the whole day\n input 2 to view the
morning\n input 3 to view the afternoon')
                message = int(input())
                quit(boats, message)
                boats = []
                for i in range(5):
                    boats.append(boat())


def main():
    welcome()

main()
```