

```
class place_infor:
    def __init__(self, place_ID, place_Name, place_feature, place_pos):
        self.ID = place_ID
        self.Name = place_Name
        self.feature = place_feature
        self.place_pos = place_pos
```

class place_infor:

- 从数据库中读取所有地点信息
- 各个要素信息
 - place_ID: 字符串型, 例如 '0001'
 - place_Name: 字符串型, 例如 'Home'
 - place_feature: 储存浮点小数的 list, 例如 [0.2, 0.2, 0.2, 0.2]
 - place_pos: 每个地点对应的 int 型 (或者浮点型) 坐标, 例如 (10, 10)

class users:

```
def __init__(self, userID, time_recoder, data_recorder, positions):
    self.ID = userID
    self.time_recoder = time_recoder
    self.data_recoder = data_recorder
    self.recoder_name = self.get_name(data_recorder, positions)
```

class users:

- 读取一个确定的 user 的信息
- 各个要素信息
 - ID: userID, 字符串型, 例如 '0000001'
 - time_recoder: 例如过去 100 天每十分钟记录的数据的对应时间, 这里对应一个 np 数组, 形状为 (100, 144, 2), 储存周几的信息 (0, 1, 2, ...6) 以及每天的时间 (int((hour * 60 + minutes) / 10))
 - data_recoder: 例如过去 100 天每十分钟记录的数据的对应坐标, 这里对应一个 np 数组, 形状为 (100, 144, 2), 储存经纬度两个数值
 - positions: 输入每个地点的坐标, 用于返回每个时间对应的地点名称

```

class recommend_system:
    def __init__(self, time_recoder, data_recoder_name):
        self.time_recoder = time_recoder
        self.local_time = time.localtime(time.time())
        self.data_recoder_name = data_recoder_name
        self.day, self.minutes = self.get_time()
        self.suggest_place = self.recommend()

```

- 用于推荐系统，读取过去的记录，来进行推荐
- 各个要素信息
 - time_recoder: 例如过去 100 天每十分钟记录的数据的对应时间，这里对应一个 np 数组，形状为 (100, 144, 2)，储存周几的信息 (0, 1, 2, ..., 6) 以及每天的时间 ($\text{int}((\text{hour} * 60 + \text{minutes}) / 10)$)
 - local_time: 从计算机系统读取时间
 - data_recoder_name: 例如过去 100 天每十分钟记录的地点信息
 - suggest_place: 返回推荐的 int 类型数值