1. 如何理解次日留存率和七日留存率[业务]

次日留存率：新登用户在首登后的次日再次登录的比例。

7 日留存率：新登用户在首登后的第七天再次登录游戏的比例。

2. 如何估算深圳市学生数量？[估算]

首先，这类估算问题会经常出现在数据分析、产品、咨询类岗位，统称为费米问题。分析这类问题可以分别从两个角度展开。根据情况，可以采用 Top-down，bottom-up 法则，即先从宏观层面，自上而下推，再由某个点横向切入，反推上去。或者也可以从需求层面和供给层面来说。然后可以对比两次推测得到的结果，如果相差不悬殊，那基本就没差啦。

然后在陈述的时候也可以需要说几句可能会出现误差的影响因素以及对结果的影响，会显得思考更加全面。具体的答案不是要求必须正确，重要的的是分析思路

这类练习题不要方，多练练思路，多看看平时的新闻报道，掌握一些基本数据 sense 就行。

对于本题可以从以下两个角度来思考~

角度一：深圳市学生数量=深圳每年高考人数*12（9 年义务制教务+3 年高中教务）+大学生数量

深圳一年的高考人数约为 5 万，有在校大学生约 10 万人

因此，深圳学生人数约为 5*12+10=70 万人。

角度二：深圳市学生数量=深圳人口*学生适龄人口比例

深圳市人口数约为 1300 万人，其中 6-22 人口占比约为 12%。因此，深圳中小学学生人数约为 1300*12%=156 万人

结果分析：根据两个角度的估算，深圳市学生数量约在 70-156 万。仍有一些因素可能导致误差，如深圳学生入学数量逐年上升，高考人数也每年增加，以高考人数来代表某一年龄的学生会产生**误**差。

3. x+y+z+m=10，均为正整数，有（）种不同的取值组合。[概统]

答案：84 种

解析：

$$x' = x - 1 \quad y' = y - 1$$
$$z' = z - 1 \quad m' = m - 1$$

则原问题转化为

$$x' + y' + z' + m' = 6$$

的非负整数解有多少个

即可以理解为四个小朋友分 6 根铅笔，有小朋友可以没有铅笔来计算 ~

即套用 重复组合 计算公式得出结论

C_6^9=84

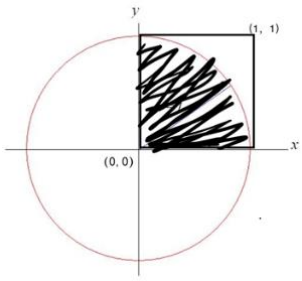
4. **在数理统计中，一般通过增加抽样次数取平均来使得预估误差减小，在机器学习中也有类似的模型处理，如随机森林，通过引入随机样本并且增加决策树的数据，对于随机森林主要降低预估的哪个方面值［机器学习算法］**

答案：预估方差
解析：这里的解题思路是 要了解随机森林的抽样方法是什么，随机森林采用袋装（Bagging）的抽样方法，而袋装的目的就是来降低方差。

5. **X 和 Y 均服从[0,1]上的均匀分布，$x^2+y^2\le 1$的概率？［概统］**

答案：$\frac{\pi}{4}$

解析：



概率即为图中阴影的面积 $\frac{\frac{\pi}{4}}{1}=\frac{\pi}{4}$

6. **简单描述特征值和特征向量［机器学习算法］**
答案：数据分析中通过计算相关 协方差矩阵的特征向量，然后确定之后的线性转换的方向。特征值表示特征向量方向转化或者压缩的强度。

7-15 题答案［SQL］

7. 每个用户消费频次：来光顾次数越多，自然用户价值越高

```
SELECT user_id, count(*) as freq from order_info group by user_id;
```

8. 消费频次的用户数分布：freq, userNum

```
select t1.freq, count(*) as userNum
from
(SELECT userid, count(*) as freq from order_info group by userid) t1
group by t1.freq
order by t1.freq;
```

9. 每个用户最近一次消费日期：最后一次消费的用户，更应被关注

```
select userid,date(max(payTime)) as lastPay
```

```
from order_info group by user_id;
```

10. 每天首次消费的用户数，目的查看付费拉新状态：firstPayDay, userNum（新用户）

```
select date, count(*) as userNum
from
(select userId,
date(min(payTime)) as date
from order_info
group by userId
) t1
group by date
order by userNum desc;
```

11. 每次消费时长的用户数分布，按小时数 group by， 输出：hourNum, userNum

```
select t1.hours,count(*) as userNum
from
(select userid, date(create_time) as date, timestampdiff(hour, createTime,payTime) as
hours from order_info) as  t1
Group by t1.hours
Order by userNum desc
```

12. 每个桌子累计订单量和用户量：deskId, orderNum, userNum

```
select t1.deskid, count(t2.orderid) as orderNum, count(distinct t2.userid) as userNum

From desk_info as t1 left join order_info as t2 on t1.deskid = t2.deskid

Group by t1.deskid

Order by orderNum desc, userNum desc;
```

13. 桌子座位数的对应的订单量，目的是看用户是喜欢单独过来还是约朋友过来：seatNum，orderNum

```
Select di.num, count(orderid) as orderNum

From order_info as o right join desk_info as di

On oi.deskid = di.deskid

Group by di.num

Order by orderNum desc, di.num desc;
```

14. 优惠券过期日期的对于用户数，目的便于到期前消费提醒， 输出：expireDate，userNum

```
Select date(expireTime) as expireDate, count(distinct userid) as userNum

From user_coupon_detail

Group by date(expireTime)

Order by usrerNum desc;
```

15. 每个用户消费总金额：这个..直接用钱来衡量了，假设优惠券类型只有折扣类型，没有满减类型

连接多个表

```
select t1.userId,

sum(case when t1.couponId>0 then price*t4.discount/100 else price end) as totalPay

from order_info t1

left join order_detail t2 on t1.orderId=t2.orderId

left join user_coupon_detail t3 on t1.couponId=t3.couponId

left join coupon_info t4 on t3.type=t4.type

left join product t5 on t2.productid=t5.productid

group by t1.userId

order by totalPay desc;
```