



量化金融分析师（AQF®）全国统一考试

模拟题

适用场次：2019 年 3 月

使用本模拟题，您应该遵守：

1. 本模拟题仅提供给参加 2019 年 3 月份 AQF 全国统一考试的考生，考生仅可以出于准备个人考试的目的查阅和打印本模拟题；
2. 严禁出于任何目的的复制、网络发布和传播、抄袭本模考题内容，如有违反，可能导致违纪或违法行为；

© 版权所有，侵权必究。

量化金融标准委员会

Standard Committee of Quantitative Finance

量化金融分析师（AQF®）全国统一考试模拟题

说明：本场考试中的代码都应采用 Python 3.X 版本作答。

1. 单选题（每题 1 分，本部分共 20 分）：只有一个正确答案，选对得 1 分，选错或不选得 0 分。

1.1. 李明，AQF，某量化基金经理，目前持有欧元/美元 1.2432 点位的多头头寸，李明认为欧元/美元汇率在 1.2400 点位存在巨大支撑，在该汇率报价跌破支撑之前不采取任何操作，而一旦欧元/美元汇率有效跌破 1.2400 点位时，该点位会从支撑线变成压力线，并且有进一步加速下跌的趋势，所以李明认为应该以 1.2395 的点位平仓多单，那么最有可能实现这个功能的交易订单是（）？

- A. 限价买单
- B. 限价卖单
- C. 止损卖单
- D. 止损买单

参考答案：C

解析：

止损卖单是指当投资者持有多单时，设立一个卖单的价格（低于当前价格），当资产价格跌破该价格时卖出资产，起到止损的效果，C 选项正确。

1.2. 多因子模型是量化交易市场上常见的交易模型。实际投资中，量化金融分析师为了得到一个满意的模型结果，需要筛选大量的因子数据，最终得到在该模型下表现良好的股票进行投资。以下关于多因子模型的说法正确的是（）？

- A. 有效因子的数量越多越好，不用担心过度拟合的问题
- B. 有效因子的筛选与当前的宏观经济趋势肯定无关
- C. 一般而言，多因子模型的因子相关性越低越好
- D. 多因子模型的经典例子不包括 Fama-French 三因子模型

参考答案：C

解析：

多因子模型中,筛选出有效因子后要对因子进行相关性分析,如果因子间存在高度相关,则需要去除重复因子或将多个因子合成为一个新因子, C 选项正确。

1.3. 李明, AQF, 某量化基金经理, 他在量化投资交易的过程中, 发现回测收益往往会大幅高于实盘收益, 最不可能造成这种情况的原因是 () ?

- A. 使用了未来函数
- B. 存在幸存者偏差
- C. 没有考虑真实交易环境限制: 如流动性不足等
- D. 在回测过程中考虑到量化策略交易成本、冲击成本等

参考答案: D

解析:

如果在回测过程中考虑量化策略的交易成本、冲击成本等, 会降低回测收益, 使得回测收益更接近于实盘收益, D 选项正确。

A 选项错误, 量化策略回测中如果使用未来函数, 会使得回测收益大幅高于实盘收益。

B 选项错误, 在回测过程中, 如果自动过滤了退市或停牌的股票, 会产生幸存者偏差, 可能导致回测收益高于实盘收益。

C 选项错误, 实盘交易时会受到真实交易环境的限制导致订单无法以回测价格执行, 从而影响实盘策略的收益。

1.4. 目前, 量化交易策略领域涌现了一大批新型的交易策略。李明, AQF, 某量化基金经理, 通过分析社交媒体上的舆情信息对股票市场走势进行研判。则该交易策略最有可能属于以下哪种类型 () ?

- A. 大数据与舆情分析
- B. 无风险套利策略
- C. 多空交易策略
- D. 事件驱动型策略

参考答案: A

解析:

随着大数据时代的到来, 大数据及舆情分析在量化投资领域获得了越来越广泛的应用, 通常使用 NLP 自然语言处理对舆情信息进行分析, 从而对股票走势进行判断, A 选项正确。

1.5. 已知 `label='close'`, `price=29.354679`, 下列代码不能输出 `'close:29.35'` 的是 () ?

- A. `"{1}:{0:.2f}".format(price, label)`
- B. `"%s:%.2f" % (label, price)`
- C. `"%s:%.2f" % (price, label)`
- D. `"{}:{:.2f}".format(label, price)`

参考答案: C

解析:

格式化字符串可以使用百分号方式或 `format` 方式, ABD 选项都可以正确输出结果, C 选项中 `price` 和 `label` 颠倒会导致报错, 因为 `label` 变量是字符串, 不是浮点型数据, 不可以进行保留两位小数的操作。

1.6. 以下关于 Python 面向对象的说法中, 正确的是 () ?

- A. 面向对象可以提升代码复用性
- B. 父类的方法不可以被子类继承
- C. 已知 B 为 A 的父类, C 为 B 的父类, 则 A 无法继承 C 的属性
- D. 父类方法不可以通过子类创建的实例对象调用

参考答案: A

解析:

面向对象可以提升代码复用性, A 选项正确。

1.7. 假设某因子截面数据以字典形式储存, 字典键为股票代码, 字典值为因子数据,

```
factor_data = {  
    '600001':10, '600002':-5, '600003':0,  
    '600004':-1, '600005':3, '600006':5,  
}
```

为选取出因子值大于 0 的股票并将股票代码储存在一个列表中, 当列表中的股票达到 2 只时正常停止选取, 则在以下横线中需要添加的代码是 () ?

```
chosen_stocks = []
```

```
for code in factor_data:
    if len(chosen_stocks)>=2:
        _____
    if factor_data[code] > 0:
        chosen_stocks.append(code)
```

- A. raise
- B. stop
- C. pass
- D. break

参考答案: D

解析:

raise 关键字, 常用在 **try...except...**或条件判断结构中, 用于抛出异常;

stop 不是 Python 关键字;

pass 是空语句, 不做任何事情, 一般用做占位语句, 是为了保持程序结构的完整性;

break 关键字, 用于循环结构中, 作用是终止并跳出循环, D 选项正确。

1.8. 李明, AQF, 某量化基金经理, 他在进行策略研究时需要从 **stock_base_data** 中提取 2018-01-03(含)至 2018-01-05(含)时间段中的股票 PE、CLOSE 数据, **stock_base_data** 为 **Pandas.DataFrame** 数据类型, 具体数据如下:

	PB	PE	ROE	CLOSE
2018-01-01	1.5	10	0.05	35
2018-01-02	1.5	10	0.05	36
2018-01-03	1.5	10	0.05	34
2018-01-04	1.4	10	-0.10	32
2018-01-05	1.4	10	-0.10	33
2018-01-06	1.4	12	-0.10	30
2018-01-07	1.4	12	-0.10	29

则李明提取数据时可以使用的代码为 () ?

- A. `stock_base_data.iloc['2018-01-03':'2018-01-05', ['PE', 'CLOSE']]`
- B. `stock_base_data.loc['2018-01-03':'2018-01-05', [1,3]]`
- C. `stock_base_data.loc['2018-01-03':'2018-01-05', ['PE', 'CLOSE']]`

D. `stock_base_data.iloc[3:6, [1,3]]`

参考答案: C

解析:

A 选项错误, `iloc` 不能进行标签索引;

B 选项错误, `loc` 不能混用标签索引和位置索引;

C 选项正确, 当通过标签索引时, 索引为双闭区间, 即 `['2018-01-03': '2018-01-05']` 可以同时取到 `'2018-01-03'` 和 `'2018-01-05'`;

D 选项错误, `DataFrame` 的索引从 0 开始, 因此 `[3:6]` 对应的时间段为 2018-01-04 至 2018-01-06, 与题意不符。

1.9. 李明, AQF, 某量化基金经理, 现将上市公司的 ROE 数据储存于 `Numpy.Ndarray` 数据结构中, 元素类型为浮点型, 并将其命名为 `roe_data`, 部分数据如下:

```
roe_data = np.array([0.02, 0.12, 0.20, 0.03, -0.04, 0.05, 0.15, 0.07, 0.18, 0.05])
```

他希望对 `roe_data` 中的元素从小到大进行排序, 可以使用的方法为 () ?

A. `roe_data.sort()`

B. `roe_data.astype(np.float)`

C. `roe_data.argmax()`

D. `roe_data.reshape()`

参考答案: A

解析:

`sort()` 方法可以对数组进行排序, A 选项正确;

`astype()` 方法可以转换数组中元素的数据类型;

`argmax()` 方法可以返回数组中最大值的索引;

`reshape()` 方法可以改变数组的形状。

1.10. 李明, AQF, 某量化基金经理, 现将某股票的基本面数据储存为 `Pandas.DataFrame` 类型, 并命名为 `stock_base_data`, 部分数据如下:

	PB	PE	ROE
stock1	1.5	13.7	NaN
stock2	3.2	NaN	0.04
stock3	2.4	16.9	0.02
stock4	NaN	60.1	0.10

李明根据以下字典数据对 `stock_base_data` 中的空值进行填充，字典的键与值分别对应需要进行填充的列与值，填充时对 `stock_base_data` 进行原地修改，则使用的代码为（）？

```
data_dict={'PB':2, 'PE':20, 'ROE':0}
```

- A. `stock_base_data.fillna(data_dict, inplace=True)`
- B. `stock_base_data.fillna(data_dict, inplace=False)`
- C. `stock_base_data.fillna(data_dict.values(), inplace=False)`
- D. `stock_base_data.fillna(data_dict.values(), inplace=True)`

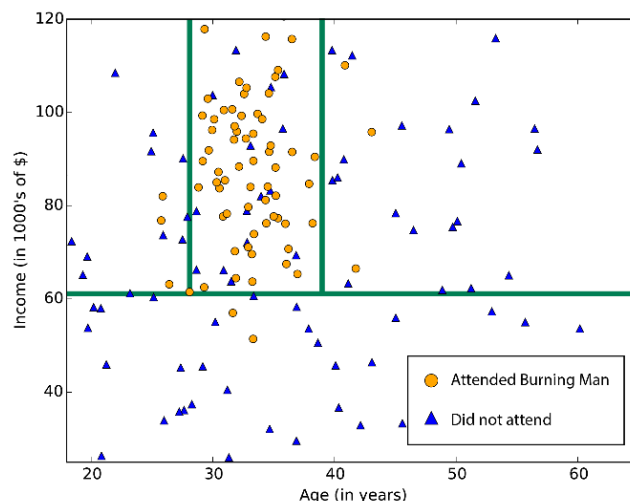
参考答案：A

解析：

本题考查缺失值的填充方法。

根据字典数据对 `DataFrame` 的空值进行填充时，需要将字典整体作为输入变量，CD 选项错误。`inplace` 参数设置为 `False` 将不会对数据进行原地修改，所以，B 选项错误，A 选项正确。

1.11. 机器学习技术目前已经越来越多地应用到量化投资领域，下图为某机器学习模型的决策边界，则该图展示的机器学习模型最有可能是（）？



- A. 逻辑回归
- B. 决策树
- C. 支持向量机
- D. 线性回归

参考答案：B

解析：

决策树模型的分类边界平行于坐标轴，因此选 B。

1.12. 一般情况下，以下哪行代码可将一个模块中的所有属性和方法一次性全部引入当前命名空间（）？

- A. `from module import *`
- B. `from module import all`
- C. `import module`
- D. `import module.*`

参考答案：A

解析：

“`from (module_name) import *`” 可以将一个模块中的所有属性和方法一次性全部引入当前命名空间，其他选项均不能达到该目的，A 选项正确。

1.13. 利用 Python 定义函数时，如果无法确定函数参数的个数，可以使用不定长参数实现。某量化研究员使用该技术编写以下函数：

```
def strategy_comment(*args):
```

```
    ...
```

则在该函数被调用并传入多个参数时，函数命名空间内的 `args` 变量名对应的数据类型是（）？

- A. 元组
- B. 列表
- C. 字典
- D. 集合

参考答案：A

解析：

使用*号将无名参数收集并传递给函数调用时，args 对应的数据类型为元组。

1.14. 李明，AQF，某量化基金经理，正在使用 Series 储存某只股票日频的交易量数据并命名为 volume_data，index 为日期，交易量部分数据如下图：

```
2018-01-28    100
2018-01-29    120
2018-01-30    310
2018-01-31    240
2018-02-01    130
2018-02-02    420
2018-02-03    350
Freq: D, dtype: int64
```

现在想降低采样频率为 5 天，并求 5 日交易量之和，则使用的代码是（）？

- A. volume_data.resample('5d').sum()
- B. volume_data.groupby('5d').sum()
- C. volume_data.groupby('5d').cumsum()
- D. volume_data.resample('5d').cumsum()

参考答案：A

解析：

时间序列的重采样操作使用 resample 方法，参数 '5d' 为重采样频率，Resampler.sum() 用于求和，A 选项正确。

1.15. 李明，AQF，某量化基金经理，正在编写策略回测代码，其中一段代码用于对 pe 值进行评价但发生了错误，代码如下：

```
pe = 20
if pe < 40
    print('The company is undervalued!')
```

则上述代码运行时，报告错误的类型与产生错误的原因因为（）？

- A. IndexError，代码索引超出范围
- B. ImportError，未导入相应模块
- C. SyntaxError，if 语句后漏写冒号
- D. SyntaxError，缩进出错

参考答案：C

解析：

题中错误为 `if` 语句后漏写冒号，属于 `SyntaxError`。

1.16. 在量化策略研究时，可以使用 Python 核心数据类型中的列表储存筛选后的股票作为股票池。现使用循环对股票池中的股票进行进一步处理，在每一次循环时需要从股票池中获取一只股票并将其从列表中删除，则可以使用的列表方法为（）？

- A. `pop()`
- B. `append()`
- C. `remove()`
- D. `extend()`

参考答案：A

解析：

使用 `pop()` 方法可以移除列表中某一位置的元素并返回该元素，A 选项正确。

`append()` 和 `extend()` 方法用于增加列表中的元素，`remove()` 方法用于删除元素，此三种方法均没有返回值。

1.17. 在量化策略回测报表输出时，某量化研究员希望对策略年化收益的显示如下图所示：

```
Annual return |  
0.13
```

现已知输出时使用的代码为：

```
print("Annual return | _____0.13")
```

则在横线处需要填写的转义字符为（）？

- A. `\\`
- B. `\000`
- C. `\n`
- D. `\t`

参考答案：C

解析：

`\n` 表示换行符，C 选项正确。

1.18. 已知 `data` 是一个 `Numpy.Ndarray` 类型数据,现需要获取 `data` 中元素的数据类型,可以使用的代码为 () ?

- A. `data.dtype`
- B. `data.size`
- C. `data.itemsize`
- D. `data.nbytes`

参考答案: A

解析:

`Ndarray` 的 `dtype` 属性可以查看数组中元素的数据类型, A 选项正确;

1.19. 代码调试是量化金融分析师必备的分析技能, 以下关于代码调试的说法中不正确的是 () ?

- A. 代码运行出现逻辑错误时, 可以通过打印中间变量的方法查找错误
- B. 大段代码运行出现错误时, 可以对代码逻辑进行分解, 逐段分析错误原因
- C. 编写复杂代码时, 必须将所有逻辑一次写完, 再对代码进行调试
- D. 复杂代码调试时可以使用 IDE 中的调试工具来提高效率

参考答案: C

解析:

C 选项, 编写复杂代码时, 一般会将复杂代码分解成几个部分, 分别编写和调试。

1.20. 李明, AQF, 某量化基金经理, 在编写量化交易策略时, 利用列表作为股票池的数据类型。现需要对某只股票是否在股票池中进行判断, 为实现该目的, 最有可能使用代码是 () ?

- A. `isin`
- B. `if`
- C. `\`
- D. `in`

参考答案: D

解析:

`in` 关键字可以判断特定的值是否在列表中存在, D 选项正确。

2. 多选题（每题 2 分，本部分共 20 分）：有 2-5 个正确答案，全部选对得 2 分，少选得 1 分，选错或不选得 0 分。

2.1. 衍生品市场如今日渐成熟，期权作为一种重要的衍生品，越来越受到市场的关注。以下关于常用期权策略的说法中，正确的是（）？

- A. 卖出期权理论上潜在收益无限，潜在风险有限，因此是常用的保护策略
- B. 期权的到期时间越长，时间价值越大，期权对到期时间的变化越敏感
- C. 预期市场中有重大事件公布时，可以通过买入跨式期权捕捉大幅价格波动
- D. 预期市场将在一段时间里处于震荡行情中，可以通过卖出跨式期权在震荡行情中获利
- E. 宽跨式期权一般比普通跨式期权的构建成本更高

参考答案：CD

解析：

A 选项，购买期权理论上收益无限，风险有限（最多损失全部的权利金），期权的卖方则相反；

B 选项，期权的剩余有效期越长，时间价值越大，对到期时间变化越不敏感；

E 选项，宽跨式期权比普通跨式更难盈利，所以构建成本相对更低。

2.2. 李明，AQF，某量化基金经理，正在研究基于统计套利的量化投资策略，以下关于统计套利的说法正确的是（）？

- A. 统计套利可能会运用一定的量化和程序化的方法进行
- B. 统计套利关注资产在统计意义上的错误定价
- C. 统计套利整体上使用的是动量延续的思想
- D. 统计套利不重视股价之间的数量关系，比如：线性相关性、协整关系等
- E. 统计套利不依赖市场趋势，因此可以一定程度上规避系统性风险

参考答案：ABE

解析：

C 选项，统计套利使用的主要是均值回归的思想；

D 选项，统计套利利用的是统计量之间的数量关系，比如线性关系，协整关系。

2.3. 动量策略和反转策略是量化投资中两大经典策略，以下哪些具体场景属于动量策略（）？

- A. 某期货品种价格连续 3 日上涨幅度超过 1%，认为该上涨趋势将延续并执行相应操作
- B. 豆油和大豆期货价格指数的比值达到历史最高点附近，认为未来该比值将回归正常水平并执行相应操作
- C. 某只股票月 K 线连续 3 期为阳线，认为未来一个月股票价格将继续上涨
- D. 某只股票月 K 线连续 3 期为阴线，认为未来一个月股票价格将出现反转
- E. 某绩优公司突然发布业绩预期减少的公告后，立刻卖出该公司股票

参考答案：AC

解析：

动量策略一般认为之前的趋势会得到延续，AC 属于动量策略。

BD 属于反转策略，E 属于事件驱动策略。

2.4. 以下使用 `lambda` 创建匿名函数的写法中，正确的是（ ）？

- A. `lambda x: abs(x)`
- B. `lambda x=-1: print(x)`
- C. `lambda x, y: if x>y: print(x)`
- D. `lambda range(x): x+=10`
- E. `lambda x+y: x, y`

参考答案：AB

解析：

考查匿名函数的用法。

`lambda` 函数的一般编写形式为：`lambda 形参: 运算逻辑`。其中，运算逻辑不能包含复杂的代码结构。

2.5. 对冲是量化投资策略中非常重要的风险管理手段，以下关于对冲的说法正确的是（ ）？

- A. 对冲有控制风险的作用，和直接平仓的效果相同
- B. 对冲有风险选择的作用，通过对冲可以实现保留特定风险的目的
- C. 投资者希望通过交易期权捕捉市场波动率扩张的行情，但不想承担基础资产价格变化的风险，则可以通过 `delta` 对冲实现以上思路
- D. 静态对冲适合于基础资产价格和对冲工具价格存在非线性关系的场景
- E. 动态对冲的成本一般大于静态对冲

参考答案：BCE

解析：

D 选项，静态对冲适合于线性关系的基础资产和对冲工具，动态对冲适合于存在非线性关系的交易品种。

2.6. 数据清洗是进行数据分析前必要的工作，则以下进行数据处理的场景中，属于异常值处理的是（）？

- A. 使用各品牌汽车历史销量数据进行汽车行业分析时，删除其中为负的数据
- B. 进行多因子策略研究时，对因子数据进行行业中性化处理
- C. 比较两类股票差异前，以某指标为标准，删除位于三倍标准差外的数据
- D. 进行机器学习前，对样本数据进行标准化
- E. 对多个数据源进行对比，分析其数据质量

参考答案：AC

解析：

- B 选项为行业中性化处理；
- D 选项为标准化处理；
- E 选项为数据质量分析。

2.7. Pandas 是量化投资中重要的数据处理工具。以下 Pandas 方法中，能够实现类似数据库合并操作的方法或函数是（）？

- A. concat 函数
- B. merge 函数
- C. joint 方法
- D. append 方法
- E. concatenate 函数

答案：ABD

解析：

在 Pandas 模块中，可用 concat 函数、merge 函数、append 方法执行合并操作。concatenate 函数属于 Numpy 模块中的拼接操作函数。joint 方法不存在。

2.8. 以下数据类型中，哪些类型可以对相同类型的数据使用‘+’号进行操作（）？

- A. 字典
- B. 整数
- C. 元组
- D. Numpy.Ndarray
- E. 浮点数

参考答案：BCDE

解析：

除字典外，整数、元组、Ndarray、浮点数均可使用‘+’号进行操作。

2.9. 在对量化策略进行回测过程中，需要规避未来数据的使用。以下操作属于使用未来数据的是（）？

- A. 使用分析师已公布的预测数据作为因子进行选股
- B. 在策略回测过程中，对正在停牌中的股票进行买卖
- C. 某技术指标计算时需要使用当前交易日前后 5 日的 K 线数据
- D. 确定回测期中间某日的因子权重时，使用整个回测期数据对权重进行寻优
- E. 进行回测前，对数据质量进行分析

参考答案：CD

解析：

C 选项计算指标时使用了当前交易日后 5 日的数据进行计算，因此属于使用了未来数据；

D 选项确定权重时，使用了当前交易日至回测期截止的数据进行寻优，因此属于使用了未来数据。

2.10. 李明，AQF，某量化基金经理，在进行期货行情研究时需要计算某商品指数的日收益率。商品指数的日收盘价数据保存为 Pandas 模块中的 DataFrame 数据类型，命名为 close_data，部分数据如下：

	close
2018-08-20	307.1
2018-08-22	310.9
2018-08-23	307.7
2018-08-24	304.8
2018-08-25	307.2

则在正常情况下，他可以使用以下哪些代码实现日收益率的计算（）？

- A. `close_data.pct_change()`
- B. `close_data/close_data.shift(1) - 1`
- C. `close_data/close_data.shift(-1) - 1`
- D. `close_data/close_data.shift() - 1`
- E. `close_data.daily_return()`

答案：ABD

解析：

日收益率计算可以使用当日收盘价/前一日收盘价 - 1 的方式进行计算。

C 选项计算时使用当日收盘价/后一日收盘价 - 1 计算日收益率；

E 选项非 DataFrame 的方法，因此选 ABD。

3. 解答题（每题 4 分，本部分总计 60 分）：按步骤、得分点给分。

3.1. 已知某天股票市场横截面数据为 DataFrame 数据类型并命名为 `stock_data`，其 `index` 为股票代码，存储数据为股票当天所有股票的成交量数据，单位为万手。现希望对所有成交量数据保留两位小数，所使用的代码是？

	VOLUME
600001	19.2448
600002	26.8842
600004	8.2433
600007	36.4546

参考答案：

`stock_data.applymap(lambda x: round(x, 2))`

或 `stock_data.round(2)`

3.2. 已知某只银行股票 2016 年 10 月 1 日至 2016 年 10 月 31 日的每日收益率数据如下：

```
{'2016/10/10':0.02, '2016/10/11':0.015, '2016/10/12':0.034,  
'2016/10/13':-0.007, '2016/10/14':-0.023, '2016/10/17':-0.059,  
'2016/10/18':0.003, '2016/10/19':0.067, '2016/10/20':0.089,  
'2016/10/21':0.010, '2016/10/24':0.003, '2016/10/25':0.001,  
'2016/10/26':-0.023, '2016/10/27':0.009, '2016/10/28':-0.058,  
'2016/10/31':0.045}
```

该数据命名为 `return_data`。请计算 5 日收益率的滚动标准差。

参考答案：

```
return_data_sr = pd.Series(return_data)  
return_data_std = return_data_sr.rolling(5).std()
```

3.3. 多因子策略是量化交易策略中最为常见的策略之一。李明，AQF，某量化基金经理，正在研究多因子选股策略，已知 `df` 为 `Pandas.DataFrame` 数据，部分数据如下图：

	LCAP	PE	REVS20
000001	25.4104	7.3031	0.9322
000002	25.1213	5.9634	0.9236
000004	20.7556	311.7743	1.0415
000005	21.4881	335.2534	0.9438
000006	22.4981	9.2826	0.8973

现在要在 `df` 中新增一列‘`mark`’，将 `REVS20` 最小的 20 只股票‘`mark`’值填充为 1，请编写相应代码。

答案：

```
index = df.sort_values('REVS20').index  
df.loc[index[:20], 'mark'] = 1
```

3.4. 某量化基金研究员正在使用 `PE` 因子进行单因子选股策略回测，请写出该策略的一种编写逻辑。（本题只需写出策略逻辑和步骤，无需代码）

参考答案：

本题为开放题，有多种策略思路，以下为简单示例：

策略步骤：

- 1) 数据获取：获取 PE 因子数据和个股收益数据；
- 2) 数据预处理：进行去极值、行业中性化等数据处理操作；
- 3) 以每月为单位，在每月月初对全市场股票 PE 数据进行从小到大排序，选出 PE 最小的 20 只股票进行等权持有，持有周期一个月，下月月初按相同逻辑进行调仓换股；
- 4) 计算策略收益，并进行策略评价，如计算最大回撤、夏普比率等。

3.5. 某研究员在进行股票分析时通过网络爬虫技术获取了 2019 年年报中全市场股票的 ROE 数据，但由于技术原因，数据储存在嵌套的列表中（其中，每个子列表中包含的元素数量相同），部分数据如下：

```
[[1.4, 3.2, 5.1],  
 [-1.6, -0.1, 1.2],  
 ...  
 [6.1, -4.0, 3.7]]
```

该数据保存在变量名 `roe_data` 中，现希望将该数据保存为 `DataFrame` 数据结构中的一列数据，列名为 `2019_roe`。保存后结果如下图：

2019_roe	
0	1.4
1	3.2
2	5.1
3	-1.6
4	-0.1
5	1.2
6	6.1
7	-4.0
8	3.7

请编写相应代码。

参考答案：

```
import numpy as np  
import pandas as pd
```

```
roe = pd.DataFrame(np.array(roe_data).flatten(), columns=['2019_roe'])
```

3.6. 某研究员在进行数据预处理时将股票代码保存为字符串，并编写了以下函数，用于判断股票代码（例如：600001）是否存在非 6 位股票代码、是否存在非数字字符两种问题，如 60##12。他编写的函数如下：

```
def check_stock_code(stock_code)
if len(stock_code) != 6:
    print('股票代码长度存在问题！')
    if not stock_CODE.isnumeric():
        print('股票代码存在非数字字符')
```

请指出该代码存在的三处错误。

参考答案：

1. 第一行末函数定义尾未写冒号；
2. 第二行代码未正常缩进；
3. 第四行 `stock_code` 误写为 `stock_CODE`。

3.7. 已知一个变量 `data`，其数据类型为 `pandas.DataFrame`，其索引为整数索引，如下图所示：

	value
0	0.255465
1	0.419060
2	0.927960
3	0.584540
4	0.267093
5	0.795627
...	...
496	0.609533
497	0.268064
498	0.472431
499	0.915447

500 rows × 1 columns

请编写代码筛选出 `data` 中索引在 100（包含）到 200（包含）之间的数据。

参考答案：

```
data[(100 <= data.index) & (data.index <= 200)]
```

3.8. 已知 `factor_data` 为 `DataFrame` 数据结构，`factor_data['factor']` 列存储了某标的的因子数据，`factor_data` 部分数据如下图：

	factor
2018-01-01	0.721584
2018-01-02	0.754876
2018-01-03	0.203021
2018-01-04	0.040109
2018-01-05	0.290914
2018-01-06	-1.019872

现要求产生交易信号保存在 `factor_data` 新建的一列上，列名为 `'signal'`，当因子值大于 2 时，信号列数据为 1，当因子值数据小于等于 2 时，信号列数据为 0。请写出实现代码（已按惯例导入 `Pandas` 和 `Numpy`）。

参考答案：

```
factor_data['signal'] = np.where(factor_data['factor']>2, 1, 0)
```

3.9. 李明，AQF，某量化基金经理，现从某数据平台调取上海证券交易所的股票数据，股票数据如下：

```
stock_data = ['10', ' ', '25', ... , '30' ]
```

在数据使用前，李明经简单检查发现少部分数据如果缺失时会在 `stock_data` 中储存为空格字符，现希望识别这些空格将其替换为 -1，并将其他数据调整为浮点型数据。请编写相应代码，将处理结果赋值给 `adjusted_stock_data` 变量。

参考答案：

```
adjusted_stock_data = []  
for data in stock_data:  
    if data.isspace():  
        adjusted_stock_data.append(-1)
```

```
else:
    data = float(data)
    adjusted_stock_data.append(data)
```

3.10. 某股票量化研究员在进行事件驱动策略研究时需要对某时间段内各只股票发布的业绩预报信息进行统计，初步统计信息汇总为一个 **DataFrame** 数据类型并命名为 **pa_data**，部分数据如下：

	code	PA
0	000001	预增
1	000002	预增
2	000004	预减

现希望对其 PA 列中信息为‘预增’的修改为 1、信息为‘预减’的修改为-1，请编写相应代码。

参考答案：

```
pa_data.replace({'预增':1, '预减':-1}, inplace=True)
```

3.11. 假设某因子截面数据以列表形式储存，分为两个列表，**stock_code** 中储存股票代码，**factor_data** 中储存该因子数据。在两个列表中，相同位置索引的股票代码和因子数据相互对应。数据如下：

```
stock_code = ['600000', '600002', '600004']
factor_data = [0.01, 0.04, 0.05]
```

现希望使用两个列表的信息创建一个字典，字典键为股票代码，字典值为股票代码对应股票的因子值，请使用字典解析编写代码。

参考答案：

```
factor_data_dict = {code:factor for code,factor in zip(stock_code,
factor_data)}
```

3.12. 李明，AQF，某量化基金经理，希望观察某日全市场股票的日收益率分布情况。已知当日股票收益率储存在一个名为 **daily_return** 的变量中，其部分数据如下：

```
{'000001':-2.67, '000002':-3.24, '000004':1.00, '000005':-0.98,
'000006':-1.24, '000007':-0.36, '000008':-0.25, '000009':1.32,}
```

请绘制当日股票收益率的频数分布直方图，并将其标题命名为“Daily return distribution”。

参考答案：

```
from matplotlib import pyplot as plt
plt.hist(daily_return.values())
plt.title('Daily return distribution')
plt.show()
```

3.13. 某研究员在进行股票筛选时，希望根据季报中某些财务因子数据，利用决策树模型筛选出问题股票（假设季报发布后，股票 3 个月收益率低于-20%为问题股票）。现将经过预处理的历史财务因子数据保存在变量名 `train_x` 中，数据类型为 `Numpy.Ndarray`，部分数据如下图：

```
array([[ 0.56828359,  0.7616555 ,  0.80349865, -0.81155623],
       [-1.45399583,  0.94687782,  1.82811314, -0.0272618 ],
       [-0.53996824, -0.87948662,  0.92591215,  0.89889853],
       [-0.99321189,  0.94756802, -0.3979266 ,  0.79418082],
       [ 1.22969928,  0.52568381, -0.46958774, -0.05550449],
```

并根据历史行情数据，将是否为问题股票的信息储存在变量名 `train_y` 中，数据类型为 `Numpy.Ndarray`，部分数据如下图：

```
array([0, 1, 0, 0, 0, 1, 0])
```

`train_x` 每行数据和 `train_y` 数据一一对应。

已知经过预处理的最新财务因子数据保存在变量名 `predict_x` 中，数据类型为 `Numpy.Ndarray`，部分数据如下图：

```
array([[ 1.58969172, -0.65247445,  1.60302934,  0.63670234],
       [ 2.11482755,  1.46192798,  0.22778077,  1.00481372],
       [-0.28243929, -1.72024605,  1.95718702, -0.03967109],
       [ 0.50257421,  0.99327239,  0.41836323, -0.39239848],
       [ 0.75388262,  0.0585765 ,  0.40982403,  1.01642748],
```

请为该研究员编写代码，利用 `train_x`, `train_y` 训练决策树模型后使用 `predict_x` 数据完成预测。

参考答案：

```
from sklearn import tree
clf = tree.DecisionTreeClassifier()
clf.fit(train_x, train_y)
```

```
result = clf.predict(predict_x)
```

3.14. 李明, AQF, 某量化基金经理, 在进行策略研究时需要了解各行业情况, 股票数据储存在变量名 `stock_data` 中, 部分数据如下图:

	industry	PE	ROE
stock1	A	10	0.02
stock2	B	20	-0.01
stock3	A	8	0.10
stock4	A	2	0.01
stock5	B	35	0.20

现要使用 `groupby` 技术:

1. 计算各行业的 PE 均值, 存储在 `pe_data` 中;
2. 并从 `pe_data` 中确定 PE 均值最小行业, 存储在 `pe_min_industry` 中;
3. 获取 PE 均值最小行业的所有股票数据, 并存储在 `pe_min_industry_data` 中。

参考答案:

```
pe_data = stock_data.groupby('industry')['PE'].mean()
pe_min_industry = pe_data.idxmin()
pe_min_industry_data =
stock_data[stock_data['industry']==pe_min_industry]
```

3.15. 已知某三只股票 2018 年 1 月 1 日至 2018 年 1 月 15 日共 12 日的日收益率数据保存在 `return_data` 中, `return_data` 的具体代码如下:

```
return_data = pd.DataFrame(
    {'stock_1':[0.01, 0.03, 0.02, 0.01, 0.05, -0.10, -0.05,-0.01, 0.01,
0.01, 0.02, 0.01],
    'stock_2':[-0.01, -0.02, 0.02, 0.01, 0.02, 0.03, 0.02, 0.05, 0.06,
0.04, 0.05, 0.07],
    'stock_3':[0.01, -0.01, 0.00, 0.01, 0.02, 0.01, -0.02, 0.00, 0.01,
0.03, -0.01, 0.00]}),
```

```
index=['2018-01-01', '2018-01-02', '2018-01-03', '2018-01-04', '2018-01-05', '2018-01-08', '2018-01-09', '2018-01-10', '2018-01-11', '2018-01-12', '2018-01-15', '2018-01-16'],  
)
```

现有某选股策略每 6 日进行一次调仓，已知该策略前 6 日等权持有 stock_1、stock_2，后 6 日等权持有 stock2、stock3，请编写代码计算策略的最终收益率。

答案：

```
return_data.loc['2018-01-01':'2018-01-08', 'portfolio_return'] =  
0.5*(return_data['stock_1']+return_data['stock_2'])  
return_data.loc['2018-01-09':'2018-01-16', 'portfolio_return'] =  
0.5*(return_data['stock_2']+return_data['stock_3'])  
final_return = (1+ return_data['portfolio_return']).prod()
```