**交易日数据**

**一、判断是否交易日**

**调用方法**

|  |
| --- |
| Python context.is\_trading\_date(date, exchange="XSHG") |

**方法说明**

|  |
| --- |
| 给定一个日期和交易所，判断该日期是否交易日 |

**参数说明**

* date: datetime或者日期字符串类型，待查询日期
* exchange: 字符串类型，交易日代码名称，默认为上海交易所XSHG

**代码示例**

|  |
| --- |
| Python def timing(context):  """择时"""  is\_trade\_date = context.is\_trading\_date("2022-02-14") |

**返回数据**

* 返回为一个布尔类型：
* True：该日期为交易日
* False: 该日期为非交易日

**二、查询下一个交易日**

**调用方法**

|  |
| --- |
| Python context.get\_next\_trading\_date(date, exchange="XSHG") |

**方法说明**

|  |
| --- |
| 给定一个日期和交易所，获取该日期的下一个交易日 |

**参数说明**

* date: datetime或者日期字符串类型，待查询日期
* exchange: 字符串类型，交易日代码名称，默认为上海交易所XSHG

**代码示例**

|  |
| --- |
| Python def timing(context):  """择时"""  next\_trade\_date = context.get\_next\_trading\_date("2022-02-14") |

**返回数据**

* 返回下一个交易日，为datetime.date类型

**三、查询上一个交易日**

**调用方法**

|  |
| --- |
| Python context.get\_pre\_trading\_date(date, exchange="XSHG") |

**方法说明**

|  |
| --- |
| 给定一个日期和交易所，获取该日期的上一个交易日 |

**参数说明**

* date: datetime或者日期字符串类型，待查询日期
* exchange: 字符串类型，交易日代码名称，默认为上海交易所XSHG

**代码示例**

|  |
| --- |
| Python def timing(context):  """择时"""  pre\_trade\_date = context.get\_pre\_trading\_date("2022-02-14") |

**返回数据**

* 返回上一个交易日，为datetime.date类型

**四、查询标的两个日期之间的交易日数**

**调用方法**

|  |
| --- |
| Python context.get\_symbol\_timedelta(symbol\_exchange, start\_time, end\_time) |

**方法说明**

|  |
| --- |
| 查询指定标的的任意两个日期之间的交易日数量 |

**参数说明**

* symbol\_exchange: 字符串类型，标的代码
* start\_time: datetime或float类型，开始时间，可以传时间戳
* end\_time: datetime或float类型，结束时间，可以传时间戳

**代码示例**

|  |
| --- |
| Python def timing(context):  """择时"""  import datetime    timedelta = context.get\_symbol\_timedelta(  "600000.XSHG",   datetime.date(2020,1,1),   datetime.date(2020,4,1)  ) |

**返回数据**

* 返回两个日期之间的交易日天数