**Binance Client Code snippets**

Rewritten Functions:

class BinanceClient:  
 def \_\_init\_\_(self, api\_key: str, api\_secret: str, testnet: bool, futures: bool):

# this function will communicate with the Binance API  
# it permits requesting, sending and deleting info when interacting with the API  
# returns the response in a JSON format, which would be a dictionary

def \_make\_request(self, method: str, endpoint: str, data: typing.Dict):

# public endpoints  
# get the possible contracts like BTC/USDT or ETH/USDT for example  
# returns a dictionary of contracts

def get\_contracts(self) -> typing.Dict[str, Contract]:

# in charge of getting the amount of all cryptos the user has  
# returns a dict of Balance

def get\_balances(self) -> typing.Dict[str, Balance]:

# manages to get the historical candlestick, up to 1000  
# receives the contract and the time interval, 1m, 5m, 15m and so on  
# returns a list of Candle

def get\_historical\_candles(self, contract: Contract, interval: str) -> typing.List[Candle]:

def place\_order(self, contract: Contract, order\_type: str, quantity: float, side: str, price=None, tif=None) -> \  
 OrderStatus: