

키움 Open API를 활용한 급등 주 추천 서비스 개발

꽂 잡아 떨어진다

201835662 박현성

201935261 박민지

201835603 강동우

201835628 김석준

201835742 조현동

목차

1. 주제 선정 배경, 개발 목표
2. 요구사항 정의 및 개발 진행
3. 소스코드 별 세부 기능 소개
4. UI를 통한 결과 확인
5. 모의 투자 분석, 출처 및 참고문헌

1. 주제 선정 배경, 개발목표

Part 1. 주제 선정 배경, 개발목표

조선경제 >

“한국에서 장기 투자? 자칫 망하는 길이에요”

신성호 전 대표는 우선 한국 증시에서 장기 투자로 큰 성과를 내기 어렵다는 점을 지적했습니다. 2000년대의 10년간 연평균 주가 상승률은 5.1%, 2010년대는 2.7%여서 서울 아파트 등 다른 자산에 투자했을 때보다 한국 증시의 투자 수익률이 낮다는 것입니다.

주식거래 늘어날까...단타 확대로 동학개미 누적손실 우려

[아시아경제 지연진 기자]내년부터 증권거래세가 인하되면서 주식거래가 활발해질 가능성이 점쳐진다. 다만 개인투자자의 단타 매매와 외국인 고빈도매매가 활발해질수 있다는 우려가 나온다.

Part 1. 주제 선정 배경, 개발목표

개발 목표

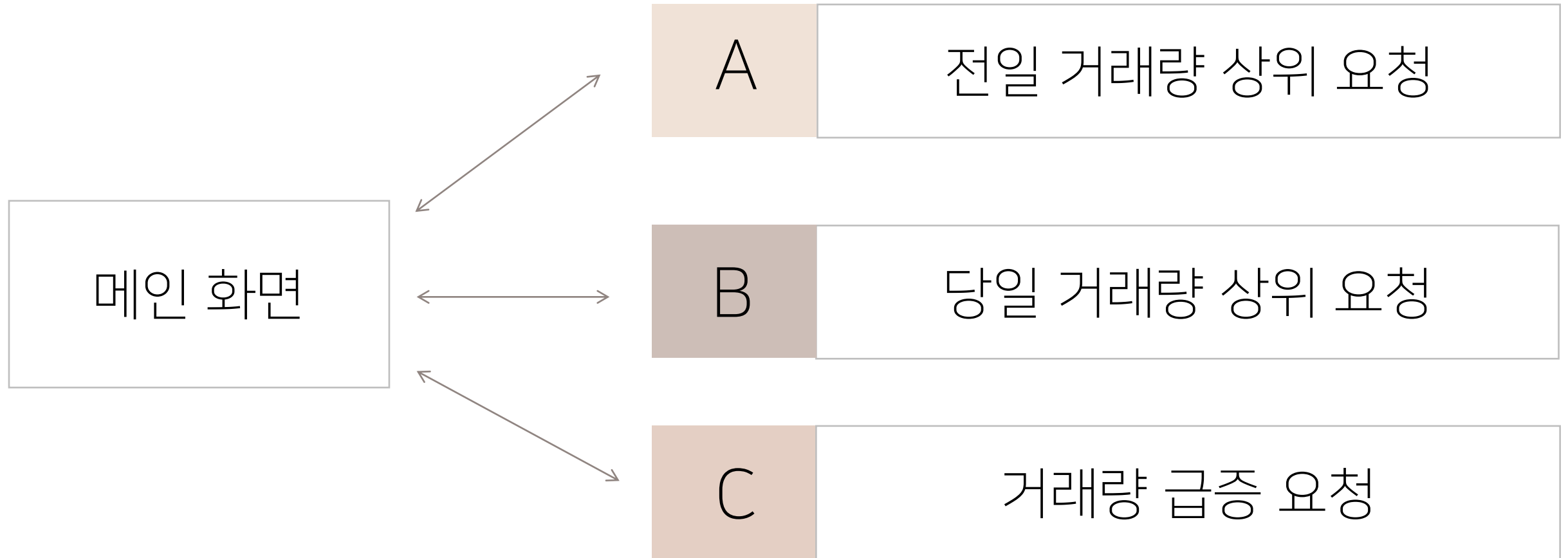
1. Kiwoom API를 이용하여 금융데이터를 다룰 수 있는 능력
2. PyQt5를 이용하여 GUI를 파이썬 환경에서 개발하는 경험
3. Qt Designer 프로그램을 활용하여 가시성 높은 UI 개발
4. UI와 API를 연결하여 사용자에게 필요한 기능만 보여주는 프로그램 개발

2. 요구사항 정의 및 개발진행

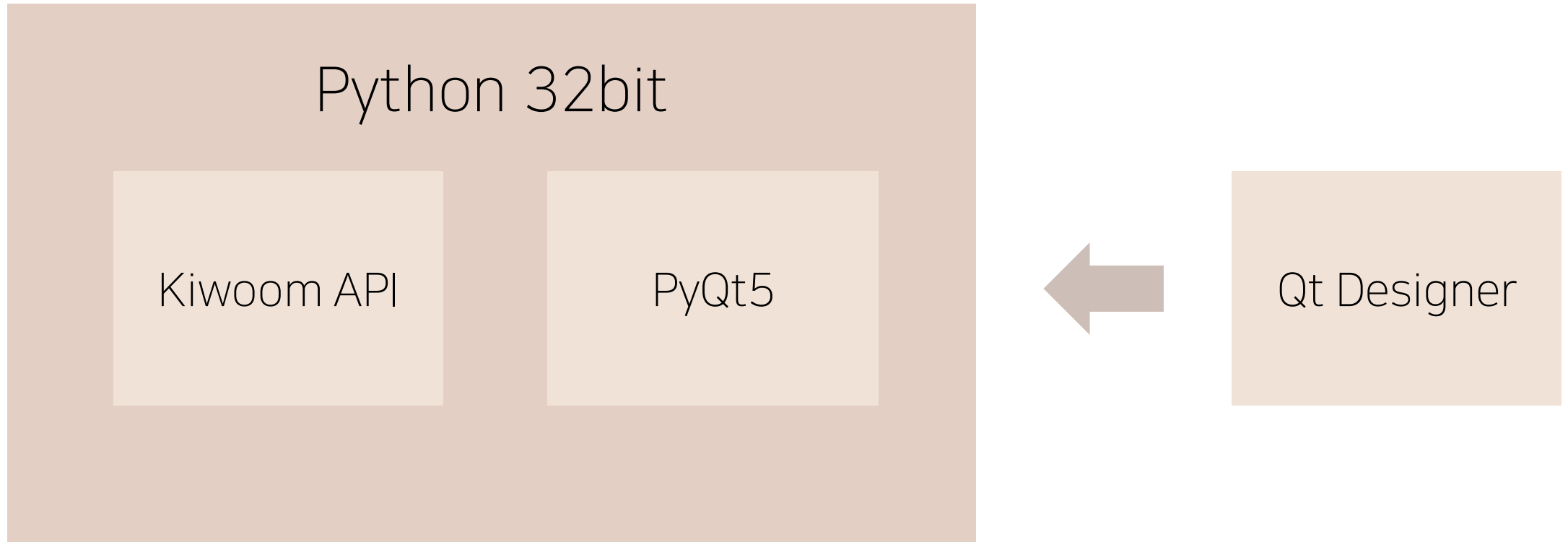
Part 2. 요구사항 정의 및 개발진행

	메인	A	B	C
기능	메인 윈도우	전일 거래량 상위 요청	당일 거래량 상위 요청	거래량 급증 요청
세부사항	기능 선택창, 타이 틀 등 위젯 포함	Tr요청	Tr요청	Tr요청 혹은 메서 드, 종목코드도 포 함
화면 흐름	메인 -> ABC	A->메인	B->메인	C->메인

Part 2. 요구사항 정의 및 개발진행



Part 2. 요구사항 정의 및 개발진행



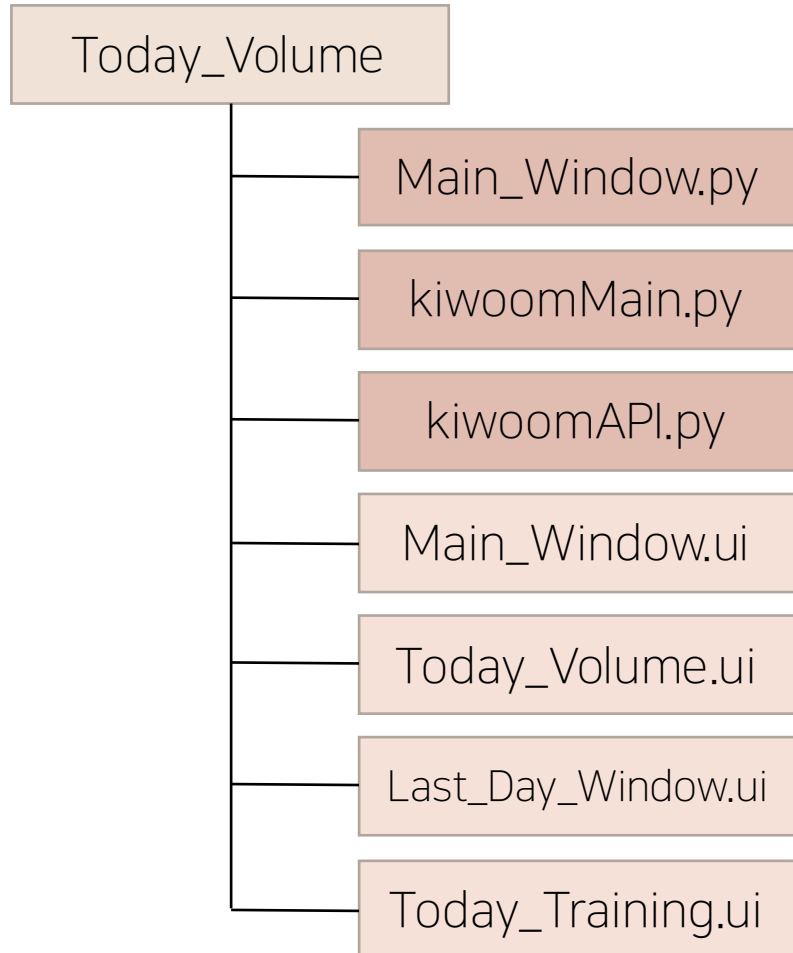
Part 2. 요구사항 정의 및 개발진행

	10/24 - 10/30	10/31 - 11/06	11/7 - 11/13	11/14 - 11/18
개발 주제 선정				
UI 및 세부 기능 개발				
PPT 작성				
최종발표				

3. 소스 별 세부 기능 소개

Part 3. 소스 별 세부 기능 소개

- 디렉토리 구성



Ui 파일 호출 및 첫 화면 출력

각 요청에 따라 어떤 값이 리턴 되는 지 정의된 함수 모음

로그인 버전처리, TR 조회 요청, 실시간 데이터 처리

Part 3. 소스 별 세부 기능 소개

- Main_Window.py

```
from PyQt5 import uic
form_main_class = uic.loadUiType("Main_Window.ui")[0]
form_Last_Day_class=uic.loadUiType("Last_Day_Window.ui")[0]
form_Today_Traiding_class=uic.loadUiType("Today_Trading.ui")[0]
Increase_Volume_class=uic.loadUiType("Increase_Volume.ui")[0]
```

UI파일과 파이썬 연결

```
#-----메인 윈도우 UI <START>-----
class Main_WindowClass(QMainWindow, form_main_class) :
    def __init__(self) :
        super().__init__()
        self.setupUi(self)
        self.username.setText(api_con.Set_User_Name())

    def Last_Day_Button_In_Main(self):
        self.hide()
        self.w=Last_Day_Class()
        self.w.exec()
        self.show()

    def Today_Traiding_Button_In_Main(self):
        self.hide()
        self.w=Today_Traiding_Class()
        self.w.exec()
        self.show()

    def Increase_Volume_Button_In_Main(self):
        self.hide()
        self.w=Today_Increase_Volume_Class()
        self.w.exec()
        self.show()

    def Test(self):
        result=api_con.OPT10019()
        print(result['Data'][0])

    def Get_login_info(self):
        print(api_con.Set_User_Name())

#-----메인 윈도우 UI <END>-----
```

메인 화면에서의 기능 정의

Part 3. 소스 별 세부 기능 소개

- Main_Window.py

```
#-----전일거래량 상위 UI <START>-----
class Last_Day_Class(QDialog, QWidget, Form_Last_Day_class):
    def __init__(self):
        super(Last_Day_Class, self).__init__()
        self.initUI()
        self.username.setText(api_con.Set_User_Name())
        self.show()

    def initUI(self):
        self.setupUi(self)

    def Fill_Data(self):
        result=api_con.OPT10031()
        self.Today_Table.horizontalHeader().setSectionResizeMode(QHeaderView.ResizeToContents)
        for i in range(10):
            self.Today_Table.setItem(i,0,QTableWidgetItem(str(result['Data'][i].get("종목명"))))

        for i in range(10):
            self.Today_Table.setItem(i,1,QTableWidgetItem(str(result['Data'][i].get("종목가"))))

        for i in range(10):
            self.Today_Table.setItem(i,2,QTableWidgetItem(str(result['Data'][i].get("전일상대매수"))))

        for i in range(10):
            self.Today_Table.setItem(i,3,QTableWidgetItem(str(result['Data'][i].get("종목수"))))

        for i in range(10):
            self.Today_Table.setItem(i,4,QTableWidgetItem(str(result['Data'][i].get("거래량"))))

        for i in range(10):
            self.Today_Table.setItem(i,5,QTableWidgetItem(str(result['Data'][i].get("거래금액"))))

    def Back_To_Main_Button_In_Last_Day_Class(self):
        self.hide()
#-----전일거래량 상위 UI <END>-----
```

전일거래량 상위화면에서의 기능

```
#-----당일거래량 상위 UI <START>-----
class Today_Trading_Class(QDialog, QWidget, Form_Today_Trading_class):
    def __init__(self):
        super(Today_Trading_Class, self).__init__()
        self.initUI()
        self.username.setText(api_con.Set_User_Name())
        self.setupUi(self)

    def initUI(self):
        self.setupUi(self)

    def Fill_Data(self):
        result=api_con.OPT10030()
        self.Today_Table.horizontalHeader().setSectionResizeMode(QHeaderView.ResizeToContents)
        for i in range(10):
            self.Today_Table.setItem(i,0,QTableWidgetItem(str(result['Data'][i].get("종목명"))))

        for i in range(10):
            self.Today_Table.setItem(i,1,QTableWidgetItem(str(result['Data'][i].get("종목가"))))

        for i in range(10):
            self.Today_Table.setItem(i,2,QTableWidgetItem(str(result['Data'][i].get("전일상대매수"))))

        for i in range(10):
            self.Today_Table.setItem(i,3,QTableWidgetItem(str(result['Data'][i].get("종목수"))))

        for i in range(10):
            self.Today_Table.setItem(i,4,QTableWidgetItem(str(result['Data'][i].get("거래량"))))

        for i in range(10):
            self.Today_Table.setItem(i,5,QTableWidgetItem(str(result['Data'][i].get("거래금액"))))

    def Back_To_Main_Button_In_Today_Trading_Class(self):
        self.hide()
#-----당일거래량 상위 UI <END>-----
```

당일거래량 상위화면에서의 기능

Part 3. 소스 별 세부 기능 소개

- Main_Window.py

```
#-----거래량 급증 UI <START>-----
class Today_Increase_Volume_Class(QDialog,QWidget,Increase_Volume_class):
    def __init__(self):
        super(Today_Increase_Volume_Class,self).__init__()
        self.initUi()
        self.username.setText(api_con.Set_User_Name())
        self.setupUi(self)

    def initUi(self):
        self.setupUi(self)

    def Back_To_Main_Window_Button_In_Today_Increase_Volume(self):
        self.hide()

    def Fill_Data(self):
        result=api_con.OPT10019()
        self.Today_Table.horizontalHeader().setSectionResizeMode(QHeaderView.ResizeToContents)
        for i in range(10):
            self.Today_Table.setItem(i,0,QTableWidgetItem(str(result['Data'][i].get("종목명"))))

        for i in range(10):
            self.Today_Table.setItem(i,1,QTableWidgetItem(str(result['Data'][i].get("현재가"))))

        for i in range(10):
            self.Today_Table.setItem(i,2,QTableWidgetItem(str(result['Data'][i].get("전일대비"))))

        for i in range(10):
            self.Today_Table.setItem(i,3,QTableWidgetItem(str(result['Data'][i].get("등락률"))))

        for i in range(10):
            self.Today_Table.setItem(i,4,QTableWidgetItem(str(result['Data'][i].get("거래량"))))

        for i in range(10):
            self.Today_Table.setItem(i,5,QTableWidgetItem(str(result['Data'][i].get("거래금액"))))
#-----거래량 급증 UI <END>-----
```

거래량 급증 화면에서의 기능

Part 3. 소스 별 세부 기능 소개

- kiwoomAPI.py

```
class kiwoomAPI(QAxWidget):
    def __init__(self):
        super().__init__()
        self.set_kiwoom_api()
        self.set_event_slot()
        self.ret_data = {}
        self.output_list = []
        self.login = 0

    # ===== #
    def set_kiwoom_api(self):
        self.setControl("KHOPENAPI.KHOpenAPICtrl.1")

    def set_event_slot(self):
        # 공통
        self.OnReceiveMsg.connect(self.E_OnReceiveMsg)

        # 로그인 버전처리
        self.OnEventConnect.connect(self.E_OnEventConnect)

        # 조회와 실시간 데이터 처리
        self.OnReceiveTrData.connect(self.E_OnReceiveTrData)
        self.OnReceiveRealData.connect(self.E_OnReceiveRealData)
```

```
# ===== #
### Event 함수 ###
## 공통 ##
def E_OnReceiveMsg(self, sScrNo, sRQName, sTrCode, sMsg):
    print(sScrNo, sRQName, sTrCode, sMsg)

## 로그인 버전처리 ##
def E_OnEventConnect(self, nErrCode):

    self.event_loop_CommConnect.exit()
    if nErrCode == 0:
        self.login = True
    else:
        self.login == False

def E_OnReceiveTrData(self, sScrNo, sRQName, sTrCode, sRecordName, sPrevNext, nDataLength, sRealData):
    print(sScrNo, sRQName, sTrCode, sRecordName, sPrevNext, nDataLength, sRealData)

    self.Call_TR(sTrCode, sRQName)

    self.event_loop_CommRqData.exit()

def E_OnReceiveRealData(self, sCode, sRealType, sRealData):
    print(sCode, sRealType, sRealData)
```


Part 3. 소스 별 세부 기능 소개

- kiwoomMain.py

```
import kiwoomAPI
```

```
def Login(self):
    self.kiwoom.CommConnect()
    return self.kiwoom.login

def Set_User_Name(self):
    user_name=self.kiwoom.GetLoginInfo("USER_NAME")
    return user_name

def getLoginInfo(self):
    loginInfo = []
    loginInfo.append(self.kiwoom.GetLoginInfo("ACCOUNT_CNT"))
    loginInfo.append(self.kiwoom.GetLoginInfo("ACCLIST"))
    loginInfo.append(self.kiwoom.GetLoginInfo("USER_ID"))
    loginInfo.append(self.kiwoom.GetLoginInfo("USER_NAME"))
    loginInfo.append(self.kiwoom.GetLoginInfo("KEY_BSECGB"))
    loginInfo.append(self.kiwoom.GetLoginInfo("GetServerGubun"))

    return loginInfo
```

```
def OPT10030(self):
    output_list={'OPT10030':['종목명','현재가','전일대비','등락률','거래량','거래금액'],}
    self.kiwoom.output_list=output_list['OPT10030']
    self.kiwoom.SetInputValue("시장구분","0")
    self.kiwoom.CommRqData("OPT10030","OPT10030",0,"0101")

    return self.kiwoom.ret_data['OPT10030']

def OPT10031(self):
    output_list={'OPT10031':['종목명','현재가','전일대비','등락률','거래량','거래금액'],}
    self.kiwoom.output_list = output_list['OPT10031']

    self.kiwoom.SetInputValue("시장구분","0")
    self.kiwoom.CommRqData("OPT10031","OPT10031",0,"0101")

    return self.kiwoom.ret_data['OPT10031']
```

4. UI를 통한 결과 확인

Part 4. UI를 통한 결과 확인

키움 Open API 로그인 창



The image shows the Kiwoom Open API login window. It features the Kiwoom logo and the text 'KIWOOM OPEN API' and '키움 Open API'. The login form includes fields for '고객 ID' (Customer ID) with a masked input '*****', '비밀번호' (Password), and '인증비밀번호' (Authentication Password). Below these are buttons for '로그인' (Login), '환경설정' (Settings), and '종료' (Exit). There are also checkboxes for '고객 아이디 저장' (Save customer ID) and '모의투자 접속' (Connect to simulation investment), both of which are checked. At the bottom, there are buttons for '공동인증' (Joint authentication), '신규가입' (New registration), and 'ID/PW찾기' (Find ID/PW), along with a note 'ID 와 비밀번호를 입력해 주십시오.' (Please enter your ID and password).

메인 윈도우



Part 4. UI를 통한 결과 확인

전일거래량 상위

	종목명	현재가	전일대비	등락률	거래량	거래금액
1	KODEX 200선물인버스2X	+2915	+40		147424662	
2	베트남개발1	-277	-7		120586345	
3	대창솔루션	+415	+40		60835907	
4	KODEX 인버스	+4750	+35		33369912	
5	스킨앤스킨	-297	-7		17819054	
6	삼성 인버스 2X WTI원유 선물 ETN	120	0		43361001	
7	KODEX 레버리지	-15290	-240		23595873	
8	KODEX 코스닥150선물인버스	+5170	+60		33060509	
9	KODEX 코스닥150레버리지	-7305	-155		17613698	
10	카카오뱅크	+27900	+300		18426232	

가져오기

가천대학교 심화 프로그래밍 짝잡아 떨어진다 조

가능 선택?
☐ RadioButton
☐ RadioButton
☐ RadioButton

돌아가기

사용자명: 강동우

전일거래량 상위 화면

- '가져오기' 버튼 클릭 시 출력
- 가장 높은 거래량을 기록한 종목 출력
- 상위 10개의 종목을 출력

Part 4. UI를 통한 결과 확인

당일거래량 상위

당일거래량 상위						
당일 거래량 상위 10 종목						
	종목명	현재가	전일대비	등락률	거래량	거래금액
1	KODEX 200선물인버스2X	+2915	+40	+1.39	147424662	424811
2	베트남개발1	-277	-7	-2.46	120586345	34576
3	대한전선	+1910	+105	+5.82	90361133	174339
4	크리스탈신소재	+1120	+255	+29.48	61699884	63750
5	대창솔루션	+415	+40	+10.67	60835907	25733
6	인디에프	-2040	-5	-0.24	50814784	110079
7	성우하이텍	+6470	+840	+14.92	49734999	314654
8	삼성 인버스 2X WTI원유 선물 ETN	120	0	0.00	43361001	4993
9	국영지앤엠	+1910	+70	+3.80	36773306	72210
10	기가레인	+1680	+385	+29.73	33775243	53370

가져오기

기능 선택?
☐ RadioButton
☐ RadioButton
☐ RadioButton

돌아가기

사용자명: 강동

당일거래량 상위 화면

- '가져오기' 버튼 클릭 시 출력
- 가장 높은 거래량을 기록한 종목 출력
- 상위 10개의 종목을 출력

Part 4. UI를 통한 결과 확인

당일거래량 급증

Form

당일거래량 급증

당일 거래량 급증 상위 10 종목

	종목명	현재가	전일대비	등락률	거래량	거래금액
1	내추럴연도핵	+3950	+910	+29.93	2351445	
2	신풍제약우	+43900	+10100	+29.88	49824	
3	태평양물산	+1675	+385	+29.84	5664879	
4	염피대산	+514	+118	+29.80	12998950	
5	경동인베스트	+119500	+27400	+29.75	506775	
6	기가레인	+1680	+385	+29.73	33775243	
7	크리스탈신소재	+1120	+255	+29.48	61699884	
8	노블엠앤비	+3470	+680	+24.37	2453980	
9	신풍제약	+27350	+4700	+20.75	4510803	
10	삼진엘앤디	+3390	+570	+20.21	32870285	

가능 선택?
☐ RadioButton
☐ RadioButton
☐ RadioButton

돌아가기

사용자 명: 강동우

가져오기

가천대학교 심화 프로그래밍 짝잡아 떨어진다 조

당일거래량 급증 화면

- '가져오기' 버튼 클릭 시 갱신
- 가장 높은 등락률을 가지는 종목부터 정렬
- 상위 10개의 종목을 출력

5. 모의투자 분석, 출처 및 참고문헌

Part 5. 모의투자 분석, 출처 및 참고문헌

	오뚜기	네이버	데브시스터즈
주요 품목	식품 제조 및 판매	IT 서비스 사업	온라인 · 모바일 게임
주요이슈 (호재, 악재)& 공시사항	<ul style="list-style-type: none">방탄소년단 진, 오뚜기 진라면 모델 발탁	<ul style="list-style-type: none">나스닥 지수 급등으로 인한 투자자 투자 심리 개선	<ul style="list-style-type: none">3분기 영업손실 38억원으 로 인한 적자 전환 (2022.11.11)4 거래일만에 8% 급등 공 매도 수량 1만2000주 (2022.11.11)
주가 (초/말)	463,000 / 444,500	22,700 / 24,850	41,600 / 49,800
수익률(%)	-10%	11%	12%

Part 5. 모의투자 분석, 출처 및 참고문헌

출처 및 참고문헌

주제선정관련 참고

<https://www.chosun.com/economy/2022/09/21/UFSCY5XY6FDI3LR6LRTQ2EZDCU/>

<https://view.asiae.co.kr/article/2022072209553598039>

개발관련 참고

<https://www3.kiwoom.com/h/customer/download/VOpenApilInfoView>

<https://jackerlab.com/kiwoom-api-tr-list/>

<https://wikidocs.net/77487>