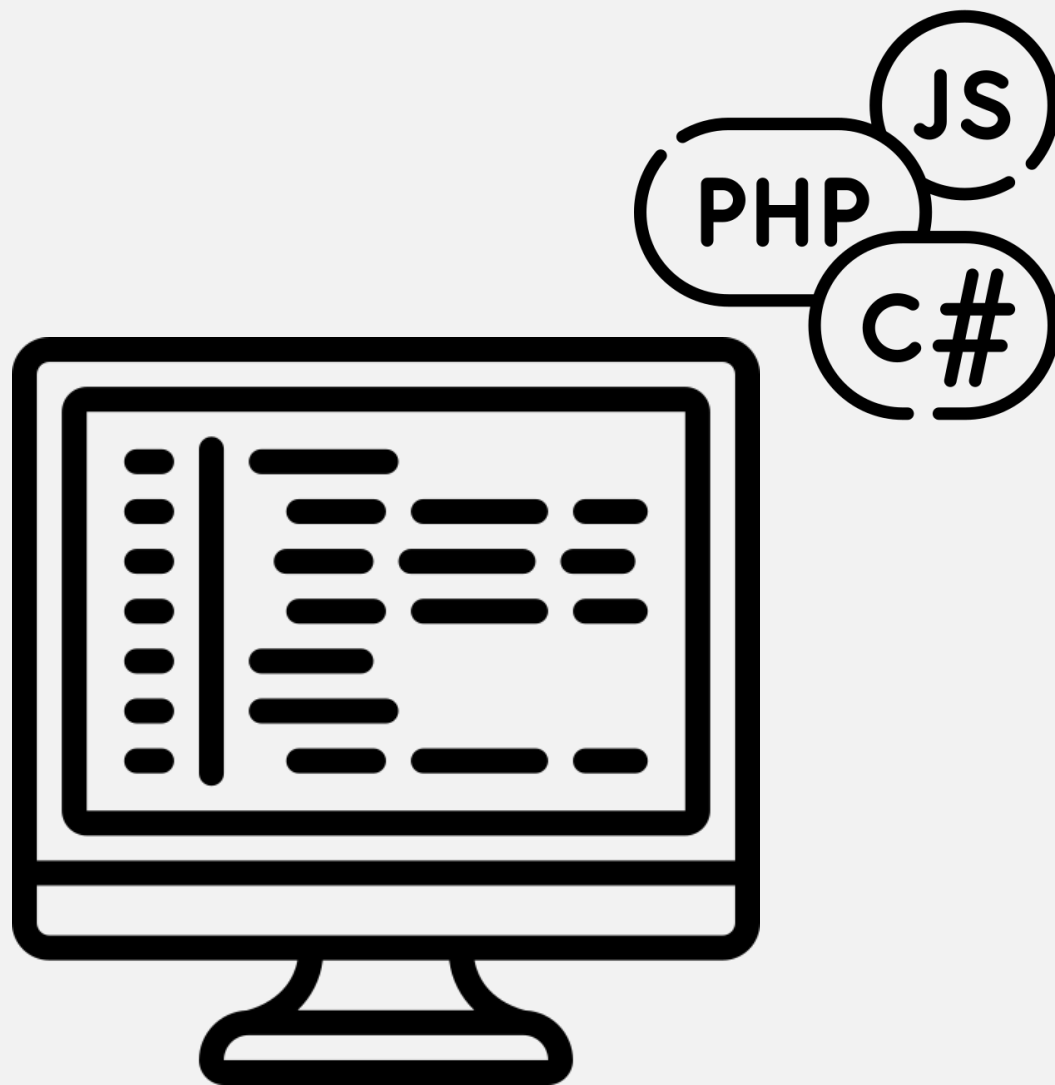


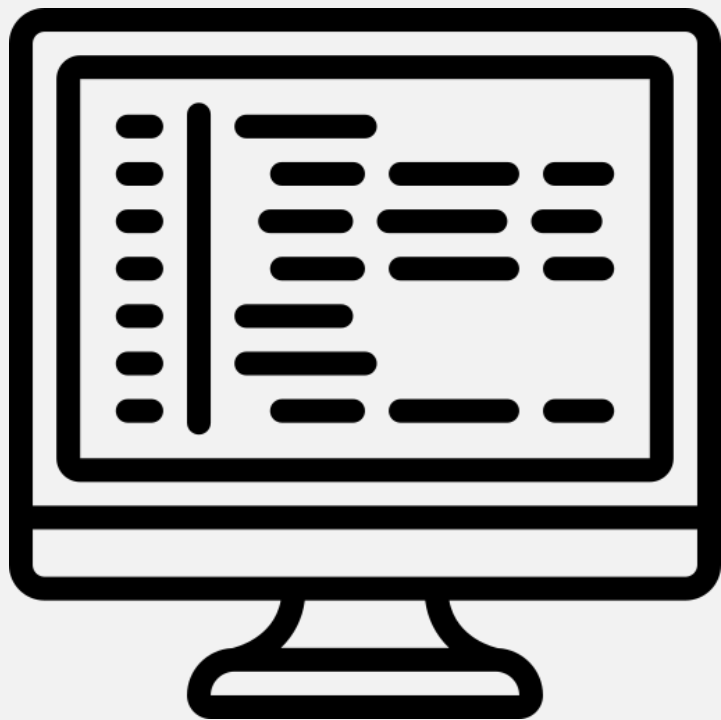
# 코딩의 기초와 문제해결

1조

신서은 이아현 최지영 최예빈



# CONTENTS



1. 코딩 주제

2. 사전 지식

3. 순서도

3. 추가한 기능

4. 해당 프로그램에 중요한 부분

5. Q&A

## 1. 코딩 주제



- 주가 그래프와 캘린더
  - 원하는 주식종목을 입력
  - 시작날짜와 종료날짜를 입력하여 기간 설정
  - 해당 주식에 대한 주가를 그래프로 나타냄
  - 해당 기간에 대한 캘린더 생성

## 2. 사전지식



- yfinance
  - 야후 파이낸스에서 크롤링한 데이터를 제공하는 라이브러리
  - 주식, 채권, 통화 및 암호 화폐에 대한 우수한 시장 데이터를 수집 가능
  - 원하는 종목의 주가 가격데이터를 받아올 수 있음
- matplotlib
  - 파이썬에서 자료를 차트나 플롯으로 시각화하는 패키지
  - 그래프를 그리도록 도와주는 라이브러리

## 2. 사전지식

- yfinance

```
pip install yfinance
```

명령 프롬프트 cmd 창에 pip install yfinance를 입력하여 설치  
➡ 설치를 해야 원하는 주식 데이터를 불러올 수 있음

```
yf.download('AAPL', start = '2019-01-01')
```

주가 데이터를 불러오는 방법은 yf를 import한 후  
원하는 종목과 시작 날짜와 종료 날짜를 입력

## 2. 사전지식

- matplotlib

```
pip install matplotlib
```

명령 프롬프트 창에 `pip install matplotlib`를 입력하여 설치  
➡ 설치를 해야 해당 데이터에 대한 그래프를 그릴 수 있음

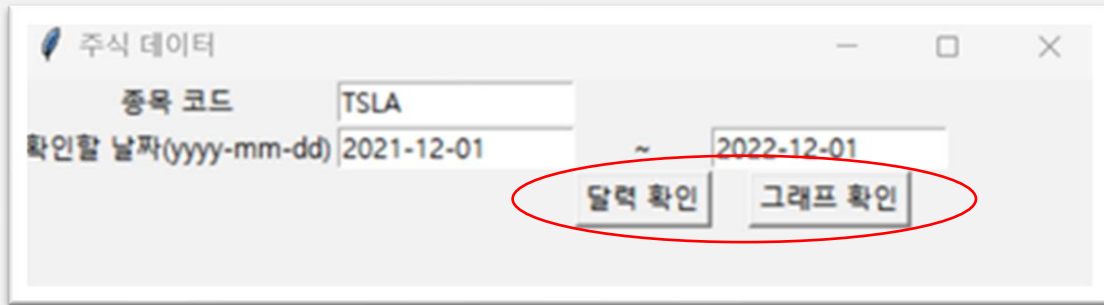
```
plt.plot(company, index, company.Close)  
plt.show()
```

그래프를 생성하는 방법은 Pyplot 모듈에 있는 '`plot()`' 함수로  
그래프를 그리고 `show()` 매소드로 그래프를 화면에 띄움

### 3. 순서도



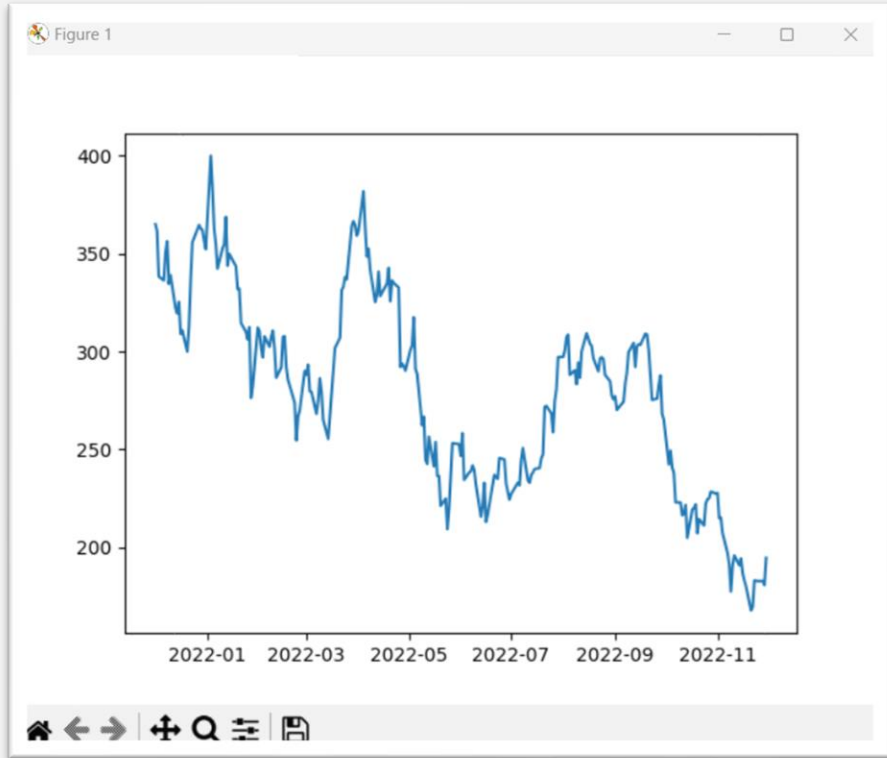
### 3. 추가한 기능



- Button 위젯 추가
  - Tkinter 모듈을 이용하여 윈도우 객체를 생성
  - 윈도우에 버튼 위젯을 내용과 함께 추가
  - 버튼을 누르면 해당 동작을 수행하게 만듦



### 3. 추가한 기능



- 주가 그래프 생성

- 그래프를 그리도록 도와주는 라이브러리: matplotlib를 불러옴
- 'def graph()' 로 주가 그래프 그리기 함수를 정의
- plt.plot으로 그래프를 그리고, plt.show로 그 그래프를 화면에 띄움
- 버튼을 누르면 해당 주식에 대한 주가 그래프가 출력

### 3. 추가한 기능



주식 데이터

종목 코드: TSLA

확인할 날짜(yyyy-mm-dd): 2021-12-01 ~ 2022-12-01

달력 확인    그래프 확인



#### • 해당 날짜의 캘린더 생성

- 확인할 날짜의 년(year), 월(month), 일(day)를 입력
- 버튼을 누르면 해당 기간에 대한 캘린더 생성
- 시작 날짜에 해당되는 달의 캘린더가 첫 페이지로 뜸
- 뒤로 넘기면서 그 이후의 날짜도 넘기면서 보는 것이 가능

## 4. 해당 프로그램에 중요한 부분

```
window = Tk()      # Tk라는 윈도우 객체 생성
window.title("주식 데이터")
window.geometry('500x100')

Label(window, text = "종목 코드").grid(column=0, row=0)
Label(window, text = "확인할 날짜(yyyy-mm-dd)").grid(column=0, row=1)
Label(window, text = "~").grid(column=2, row=1)

stock = Entry(window, width=15)      # 주식 '종목코드' 입력칸 생성
start_day = Entry(window, width=15) # '시작 날짜' 입력칸 생성
finish_day = Entry(window, width=15) # '끝 날짜' 입력칸 생성
stock.grid(column=1, row=0)          # '종목코드' 1행 0열 위치
start_day.grid(column=1, row=1)      # '시작날짜' 1행 1열 위치
finish_day.grid(column=3, row=1)     # '끝 날짜' 3행 1열 위치
```

### 소스 코드

‘window=Tk()’ 로 Tk라는 윈도우 객체를 생성

window의 제목을 ‘주식 데이터’로 설정

‘Label(window, text)’를 통해 레이블 객체를 생성하고 레이블에 띄울 내용 입력

‘Entry()’로 문자나 숫자가 입력 가능한 칸을 생성

엔트리에 주식 종목, 시작날짜, 종료날짜를 입력 할 수 있게 함

## 4. 해당 프로그램에 중요한 부분

```
# '달력' 확인 버튼 생성
btn_graph=Button(window,text="달력 확인" )
# '그래프' 확인 버튼 생성
btn_calendar=Button(window,text="그래프 확인",command = graph)
btn_graph.grid(column = 2, row = 3)
btn_calendar.grid(column = 3, row = 3)
window.mainloop()
```

### 소스 코드

Tkinter라는 모듈을 이용하여 'btn=Button'을 통해 윈도우에 버튼 위젯을 생성

'Text=' 에 '달력 확인'과 '그래프 확인'이라는 내용을 입력하여 달력 확인 버튼과 그래프 확인 버튼 생성

'달력 확인' 버튼을 누르면 원하는 날짜에 대한 캘린더가 생성되어 화면에 뜸

'그래프 확인'버튼을 누르면 해당 주식에 대한 주가 그래프가 출력

## 4. 해당 프로그램에 중요한 부분

### 소스 코드

‘def calender’ 로 달력 표시 함수를 정의

‘split(-)’ 함수를 통해 날짜를 -를 기준으로 연(year), 월(month), 일(day)을 나눠줌

입력한 년, 월, 일을 ‘int’를 통해 정수형으로 받은 후, 이를 datetime 모듈에 넣기

받아온 값으로 캘린더를 생성하고 ‘cal.pack()’ 을 통해 캘린더를 화면에 위치시킴

```
def calendar():    # 추가변동 달력 표시함수
    root = Tk()
    root.title("Calendar")
    root.geometry("600x400")

    str_date = start_day.get() #get으로 값 불러오기
    str_year, str_month, str_day = str_date.split('-')
    # -를 기준으로 문자열을 잘라주는 split
    str_date = datetime.date(int(str_year),int(str_month),int(str_day))
    #입력한 year,month,day를 int로 문자열에서 정수로 변경 이를 datetime 모듈에 넣기

    cal = Calendar(root, selectmode="day", year=str_date.year, month=str_date.month, day=str_date.day)
    #달력 생성
    cal.pack(fill = "both", expand = True)

    root.mainloop() #이벤트루프 생성
```

## 4. 해당 프로그램에 중요한 부분

### 소스 코드

```
import matplotlib.pyplot as plt
# 그래프를 그리도록 도와주는 라이브러리: matplotlib
```

```
def graph():          # 주가 그래프 그리기 함수
    print('특정 기간내의 [주가] 데이터 받아오기')
    company = yf.download(stock.get(), start_day.get(), finish_day.get())
    # company: 원하는 회사의 주가 데이터 담기
    print(company)
    plt.plot(company.index, company.Close) # plt.plot으로 그래프를 그리고,
    plt.show() # plt.show로 그 그래프를 화면에 띄운다.
```

matplotlib 패키지에 그래프를 그리는 'pyplot' 모듈을 사용하기 위해 import시킴

'def graph' 로 딜렉 표시 함수를 정의

'company'에 yfinance라이브러리를 이용하여 원하는 회사의 주가 데이터를 담음

Pyplot 모듈에 'plot()'함수를 이용하여 주가 데이터에 관해 그래프를 그리고 show() 매소드를 호출하여 그래프를 화면에 띄움

