

02

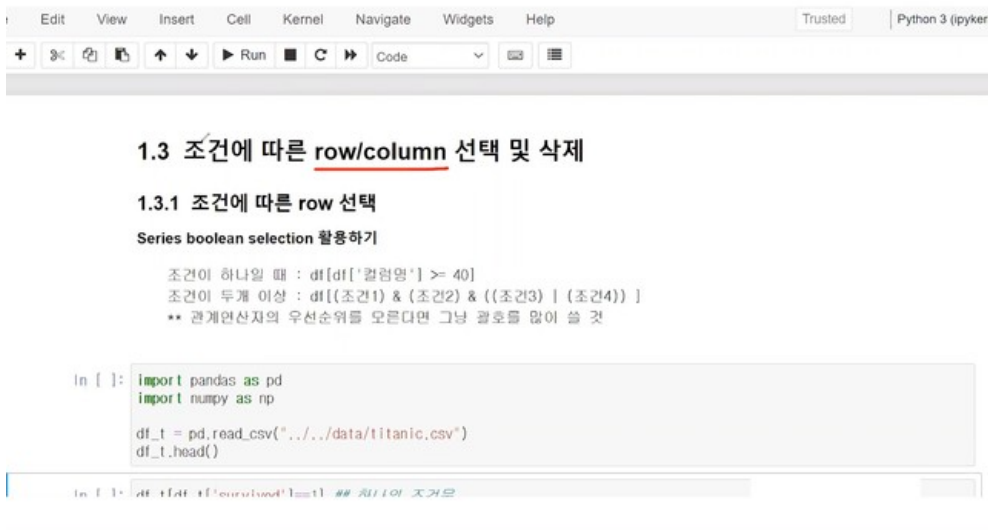
## 조건에 따른 행, 열 선택 (loc,iloc)

[조건에 따른 row 선택]

00:08

# Series boolean selection 이해가 중요합니다.

조건문으로 True 와 False 시리즈를 만들 수 있습니다.



[조건에 따른 columns 추가]

05:13

# `loc[ 조건문 , 선택할 컬럼명 ]` 을 기억해주세요!!

# `loc`를 쓰지 않고 조회한 데이터프레임에 값을 대입하면 복사본에 저장을 하므로 오류가 나요. 원본 데이터를 변경할 때에는 꼭 `loc`나 `iloc`를 써야합니다.

```

Edit View Insert Cell Kernel Navigate Widgets Help Trusted Python 3 (ipykernel)
+ < > Run Code
889 1 1 male 26.0 0 0 30.0000 C First True Cherbourg
396 rows x 11 columns

1.3.2 조건에 따른 column 추가, 변경

In [ ]: df_t.loc[df_t["survived"]==1, 'temp'] = 99999
df_t
I

In [ ]: df_t['temp2'] = False
df_t.loc[df_t.age>30, 'temp2']

In [ ]: df_t.loc[df_t.age>30, 'temp2'] = "30대 이상입니다."

In [ ]: df_t

In [ ]: ## 데이터 프레임으로 접근하여 값을 대입하면 오류가 남 // 데이터 프레임 접근은 복사본이므로 데이터에 직접 접근할
df_t[df_t["age"]>30]['temp2'] = True

```

[numpy를 활용한 조건문 활용]

08:30

np.where() 보다는 np.selection() 활용 방법을 기억해주세요

범주형 변수를 만들 때, 많이 사용합니다.

하지만 loc를 활용하여 여러 번 반복하면 selection도 하지 않아도 되겠죠??

간단한 것은 loc, 많은 조건이 있는 경우에는 selection을 사용하시면 편합니다

```

Edit View Insert Cell Kernel Navigate Widgets Help Trusted Python 3 (ipykernel)
+ < > Run Markdown
Name: temp2, Length: 305, dtype: object

In [19]: df_t.drop(columns=['temp', 'temp2'], inplace=True)

1.3.2.1 np.where()
• 데이터 프레임 or 리스트 내 필드에 접근하고 수정하는 방법. (loc or iloc를 두 번 사용하는 것과 동일)
• np.where(condition, x, y)는 조건이 충족되면x를, 그렇지 않으면y를 반환합니다.

In [ ]: df_t['alive'] = np.where(df_t['survived']==1, 'Yes', 'No')
df_t.head(10)

1.3.2.2 np.select()
• 데이터 프레임 or 리스트 내 필드에 접근하고 수정하는 방법.
• 조건 목록과 선택 목록을 입력으로 받아서 조건에 따라 선택 목록의 요소로 구성된 배열을 반환합니다.
• 이 방법을 사용하면 두 개 이상의 조건이 있을 때 Pandas의 주어진 조건을 기반으로 DataFrame 열을 만들 수 있습니다.

```

다음 시간에는 데이터를 정렬하는 방법에 대해 배워보겠습니다

다음 강의에서 뵈요~