Topic	Data Science 4	
Class Description	**Students will be comparing the habitable planets with Mars and would also be learning about more factors which make a planet habitable for us to survive.**	
Class	**C134**	
Class time	**45 mins**	
Goal	• Merging the lists to get the final list of habitable planets • Comparing with Mars • Learning about more factors that makes a planet habitable.	

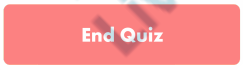
Resources Required	• Teacher Resources
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen
	• Student Resources
	  ○ Laptop with internet connectivity
	  ○ Earphones with mic
	  ○ Notebook and pen	

Class structure	**Warm Up** **Teacher-led Activity** **Student-led Activity** **Wrap up**	**5 mins** **15 min** **15 min** **5 min**

CONTEXT

- **Review the concepts learned in the earlier classes**

Class Steps	Teacher Action	Student Action
Step 1: Warm Up (5 mins)	Hi <Student Name>. In the last class, we learned about the speed of a planet! Can you tell me the formula of it?	**ESR:** - Speed of a planet is the distance covered by it divided by the time taken by

		the planet to cover the distance.
	I have an exciting quiz question for you! Are you ready to answer this question? Teacher click on the  button on the bottom right corner of your screen to start the In-Class Quiz. A quiz will be visible to both you and the student. Encourage the student to answer the quiz question. The student may choose the wrong option, help the student to think correctly about the question and then answer again. After the student selects the correct option, the  button will start appearing on your screen. Click the End quiz to close the quiz pop-up and continue the class.	
	Awesome! In today's class, we will be finding out the final list of habitable planets (if any) and we will learn about planets in our solar system. We will also be learning about more factors which makes a planet habitable.	**ESR:** "Yes!"

	Are you excited?	
	Let's get started!	
Teacher Initiates Screen Share		
CHALLENGE • **Merging the two lists** • **Comparing these data points with that of the planets in our solar system**		
Step 2: Teacher-led Activity (15 min)	*(Before beginning the class, make sure to use the same colab that you used in the last class. This is the continuation of that.)*	
	In the last class, we came up with 2 lists - **goldilock_planets and speed_supporting_planets.**	

```
[ ] goldilock_planets = list(suitable_planets) #We will leave suitable planet list as it is

    temp_goldilock_planets = list(suitable_planets)
    for planet_data in temp_goldilock_planets:
      if planet_data[8] < 0.38 or planet_data[8] > 2:
        goldilock_planets.remove(planet_data)

    print(len(suitable_planets))
    print(len(goldilock_planets))

    696
    25
```

```
[ ] planet_speeds = []
    for planet_data in suitable_planets:
      distance = 2 * 3.14 * (planet_data[8] * 1.496e+9)
      time = planet_data[9] * 86400
      speed = distance / time
      planet_speeds.append(speed)

    speed_supporting_planets = list(suitable_planets) #We will leave suitable planet list as it is

    temp_speed_supporting_planets = list(suitable_planets)
    for index, planet_data in enumerate(temp_speed_supporting_planets):
      if planet_speeds[index] > 200:
        speed_supporting_planets.remove(planet_data)

    print(len(speed_supporting_planets))

    8
```

	Do you remember what **goldilock_planets** were?	**ESR:** They were the planets that reside in the habitable zone

		of their solar system (0.38AU to 2AU).

	Awesome. Now, that we have the two lists, let's merge them together to get the list of final planets that we could get based on the data that are habitable.
	
	`habitable_planets = []`
	`for planet in speed_supporting_planets:`
	`if planet in goldilock_planets:`
	`habitable_planets.append(planet)`
	`print(len(habitable_planets))`
	
	Here, we are creating a new list, habitable_planets.
	
	We are iterating over all the planets that support speed and we are checking if this planet is in goldilock planets or not. If it is, we are appending this planet into our habitable_planets list.
		ESR:
		Using Game States

	We found 6 such habitable planets! Awesome! But the question is, are these planets really habitable? We don't know yet! Let's see an example. In our solar system, we know that the first 4 planets **Mercury, Venus, Earth and Mars** are **Terrestrial Planets!** Rest all the planets are either Gas Giants or Neptune-Like (and we do not have Super Earth in our Solar System). Thus, we can only look at our own planets. The first thing we filtered out is Gravity: **Mercury - 3.7m/s** **Venus - 8.87m/s** **Earth - 9.8m/s** **Mars - 3.8m/s** This makes all these planets habitable. Similarly, since we have these planets as terrestrial planets, we are still good to go! Now if we talk about their position in the solar system;	

	Mercury - 0.4AU **Venus - 0.7AU** **Earth - 1AU** **Mars - 1.5AU** This again, makes all these planets lie in the Goldilock zone! Now, if we talk about the speed of these planets, our final filter then: **Mercury - 47km/s** **Venus - 35km/s** **Earth - 30km/s** **Mars - 24km/s** Again, this makes all these planets habitable! Despite that, we are only considering the possibility to go to Mars and colonize it, instead of going to the other planets! This means that there are countless other reasons. Do you know a few of them?	**ESR:** Varied
	Let's get into the reasons for our own planets: **Mercury -** Mercury is not habitable since it does not have an atmosphere and its temperature varies from 100 Degree Celsius to 700 Degree Celsius. **Venus -**	

	Venus is an extreme planet and it's very hot. It's atmosphere traps the heat on the surface and its temperature is a whopping 700 Degree Celsius. There are also rains of sulphuric acid. **Earth -** Seems like just the right planet to exist. **Mars -** Mars has some atmosphere and it also has water on its surface. The temperatures are a bit extreme (20 Degree to -150 degree) but it is still manageable compared to others. The extreme temperatures are a result of the distance from the sun. Now, since we are saying that the Goldilock zone could be anywhere from 0.38 to 2 AU, and both Mercury and Venus lie in that zone, why are they so hot?	**ESR:** varied
	These planets are hot because goldilock zone is different for different stars. It depends on how big and powerful the star is, for us, we are in the beginning of our Solar System's Goldilock zone and Mars is at the very end of it. To find exact goldilock zones for different starts is a bit difficult and there's no such formula as yet, but the	

	solar system having 0.38AU as goldilock zone would be much smaller than our sun and the solar system having goldilock zone higher than 1.5AU (could also be greater than 2, up to 10) are the stars that are much more bigger and powerful than our sun! Apart from this, we also have planets tidally locked to their stars. Do you know what that means?	**ESR:** Only one side faces the sun. The planet does not rotate.
	That's right. We have day and night cycles here, which means that we are not tidally locked. You may have experienced that the days are warmer and nights are cold! That's because we are facing towards the sun during the day and facing away from the sun during the nights. When a planet is tidally locked, one side of the planet might be extremely hot while the other side of the planet might be extremely cold, which makes it impossible for us to exist. For reference, our moon is tidally locked to our planet Earth. We can always only see one side of the moon and the other side of the moon is never visible to us. We don't know what might lie there!	
Teacher Stops Screen Share		

	Now it's your turn. Please share your screen with me.	
• **Ask Student to press ESC key to come back to panel** • **Guide Student to start Screen Share** • **Teacher gets into Fullscreen**		
ACTIVITY • **Student code to segregate the lists of different planets!**		
Step 3: Student-Led Activity (15 min)	Now that we have performed our analysis and learned about a lot of concepts of science and space, it's time for us to start working on the app! We will be making a mobile app where we display all the planets and provide stats about the planet, why it's suitable, why it isn't, etc. For that, let's revisit all the steps we have done so far. Can you tell me about it?	**ESR:** • We filtered out planets based on their gravity. • We then filtered out planets based on their type. • We then filtered out planets based on whether they are in the Goldilock Zone or not and if their speeds support us humans. • We prepared the final list of planets.
	Great! Now let's reverse engineer our data and prepare 5 lists that contains all the data: 1. Planets that support only Gravity. 2. Planets that support both gravity and are of suited types. 3. Planets that support all gravity, are of suitable type and fall in goldilock zone.	*Students write code to create a red square sprite and control it using right and left arrow keys.*

	4. Planets that support all gravity, are of suitable types and have suitable speed that we can survive.
	5. Planets that are potentially habitable that support all the above filters.
	
	Now, you may say that we already have such lists, but if we just take the list of planets that support gravity (3,951) but this also contains the planets that are of suitable planet types in this. We only need those planets that just support gravity and nothing else.
	
	We need to ensure that the sum of all the planet lists comes out to be 4,250 (since we filtered out a few planets early on from here), which is the exact number of planets we started with.
	
	We need to iterate over all the planets and see what features they support. We can maintain a dictionary of all the features supported by a planet.
	
	`final_dict = {}`
	
	`for index, planet_data in enumerate(planet_data_rows):`
	`features_list = []`
	`gravity = (float(planet_data[3])*5.972e+24) /`
		Help the student and understand and code for the same.

```
(float(planet_data[7])*float(pla
net_data[7])*6371000*6371000) *
6.674e-11
 try:
   if gravity < 100:

features_list.append("gravity")
 except: pass
 try:
   if planet_data[6].lower() ==
"terrestrial" or
planet_data[6].lower() == "super
earth":

features_list.append("planet_typ
e")
 except: pass
 try:
   if planet_data[8] > 0.38 or
planet_data[8] < 2:

features_list.append("goldilock"
)
 except: pass
 try:
   distance = 2 * 3.14 *
(planet_data[8] * 1.496e+9)
   time = planet_data[9] * 86400
   speed = distance / time
   if speed < 200:

features_list.append("speed")
 except: pass
 final_dict[index] =
features_list
```

```
print(final_dict)
```

Here, we are creating a final dict, and iterating over our original **planet_data_rows** and using the enumerate function.

We are then creating a features list where we want to keep track of all the features.

We are then calculating the gravity with the same formula as above.

We are then trying to apply all the filters we have done so far inside the try and except blocks! The reason why we used try and except here is to handle any forms of errors that may come in comparing two values, etc.

Finally, we are adding this to the dictionary with the features as the value and the index of the planet as the key.

Now, we have a dictionary that contains the index of all the planet_data as key and the values are the features of this planet that can help support life!

```
final_dict = {}

for index, planet_data in enumerate(planet_data_rows):
  features_list = []
  gravity = (float(planet_data[3])*5.972e+24) / (float(planet_data[7])*float(planet_data[7])*6371000*6371000) * 6.674e-11
  try:
    if gravity < 100:
      features_list.append("gravity")
  except: pass
  try:
    if planet_data[6].lower() == "terrestrial" or planet_data[6].lower() == "super earth":
      features_list.append("planet_type")
  except: pass
  try:
    if planet_data[8] > 0.38 or planet_data[8] < 2:
      features_list.append("goldilock")
  except: pass
  try:
    distance = 2 * 3.14 * (planet_data[8] * 1.496e+9)
    time = planet_data[9] * 86400
    speed = distance / time
    if speed < 200:
      features_list.append("speed")
  except: pass
  final_dict[index] = features_list

print(final_dict)
```

```
{0: [], 1: [], 2: ['gravity'], 3: ['gravity'], 4: ['gravity'], 5: [], 6: ['gravity'], 7: ['gravity'], 8: ['gravity'], 9: ['gravity'], 10: [], 11: [], 12: [],
```

Teacher Guides Student to Stop Screen Share

FEEDBACK

- **Appreciate the student for their efforts**
- **Identify 2 strengths and 1 area of progress for the student**

	So, in this class, we merged and came up with a final list of features for all the planets that we can use for our Flask API. We also learned about some other factors that are crucial in determining if the planet is habitable or not (with reference from our own solar system). How was your experience?	**ESR:** Varied
	Amazing. While working on this project, we also made sure that we are at the top of all the concepts we have acquired so far. In the next class, we will cross-verify this dictionary's data with the data we	-

	found out and then create a Flask API for the same, so we can create a mobile app with a catalog of the analysis and data for all the exo-planets!	
Teacher Clicks 		

Activity	Activity Name	Links
Teacher Activity 1	Solution	https://colab.research.google.com/gist/shubhamwhj/0bba39505430289408ecdb6c4b12dec4/copy-of-pro-c134-reference-code.ipynb