Session 12 Info/QA

Problem Solving and Research  
Session 12 on 5/12/2020 at 5:10PM

## Research

It is important in Game Development to have a sense of what you enjoy in other games, and how do games that are already out there compare to the game that you are making. You should ask yourself:  
What do I like about the other games out there?   
What features do they have?   
What features do I really like in this game?   
How can I implement the features I like into my game?   
What can I change in these features to make them my own?

Games are never entirely created based on someone’s creativity alone. Their creativity comes from their prior experience from other games that they have played because people will always create things that are inspired from things they know. It is not plagiarising or stealing if you take a feature of another game and adapt it to your own. However, to ensure that you are not just blatantly copying that feature, it is important to change it a little bit and add something to it to make it your own.

For example, in the game Minecraft created by ‘Notch’ (Markus Persson), many different games were used to create the design, ideas and inspiration for Minecraft. Notch found his inspiration from other games and forums. His games, and Minecraft, were inspired by games like Dwarf Fortress, Roller Coaster Tycoon, Grand Theft Auto: Chinatown Wars. It is also good to adapt ideas that you created in your previous works. This was shown in Notch’s early works, such as Infiniminer, which was a block based open-ended mining game released in 2009. Infiniminer was eventually scrapped by Notch but it was the precursor of Minecraft, bringing to it the visual style, the block building, and the ‘blocky’ style! Notch incorporated different RPG elements from games that he liked and joined the, with an improved version of Infiniminer to create Minecraft.

### Problem Solving

It is important to understand the problem-solving process when creating algorithms and features. The first step in this process is to undergo a process called testing.

Testing is when you ‘test’ a feature/algorithm by providing it with many different tasks (or test-cases) to complete! For example, if you had a number guessing game that you needed to test you would test it by first putting in some valid inputs, such as 10,20,30 etc. If that worked then you can move on to the next case, which should not work but the game should ask you to retype the number, these cases could be something like letters, symbols, or negative cases. *In short, you test a feature by providing it with many scenarios designed to test if the feature does what it was intended to do and with scenarios that could possibly break it*.

When you first make a feature, it is important to do thorough testing of it. Once you have tested this feature, you should spend the time adding or creating any ‘fixes’ to your feature if the testing wasn’t completed perfectly. You may not catch all the errors in the testing, so it is important you continually revise the feature when required.

## Player Preferences

You can save basic options, scores and other data using the Player Preferences system in Unity. It is noted that you should not be using this to save data from levels as it can be easily modified by the player. In this case it is used as an example on how to save data.

For more information visit: <https://docs.unity3d.com/ScriptReference/PlayerPrefs.html>

# Questions!

What are some games that you really like?

|  |
| --- |
|  |

Why do you enjoy these games?

|  |
| --- |
|  |

What are 3 features from these games that you enjoy?

|  |  |
| --- | --- |
| 1 |  |
| 2 |  |
| 3 |  |

How can you adapt these features to make them your own?

|  |
| --- |
|  |

If you want, how could you add these features to your own game?

|  |
| --- |
|  |