Lab 4 – Objects

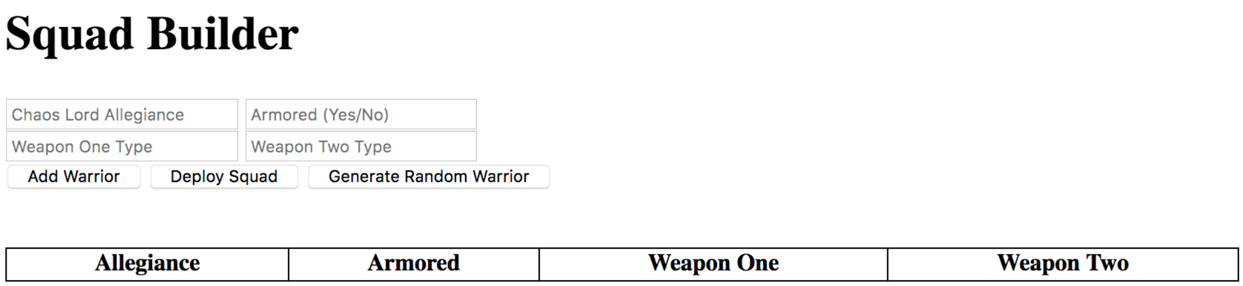
The goal of this lab is to improve your understanding of functions, objects, and general JavaScript usage. **Parts 1 and 2 of this lab are required**. **Part 3** of this lab is optional for **bonus points.**

Part 1 (Required)

1. Create a form that will allow a user to deploy a squad by creating several warriors. The user will type warrior information into multiple input boxes to create warriors**. A warrior has 3 properties**:

    - allegiance – the chaos lord this warrior follows
    - armored – whether the warrior is wearing armor or not
    - weapons – a warrior may have 1 or 2 weapons.

2. Your HTML file should be named **objects-lab.html** and should contain
    - Four text input boxes
        - Chaos Lord Allegiance
        - Armored
        - Weapon 1 Type
        - Weapon 2 Type
    - Two buttons (three if you do the bonus)
        - Add Warrior
        - Deploy Squad
    - A table with a defined header row (see below)
    - Here is an example of what the HTML should display. This one already has some CSS added to the table to show borders. Yours does not have to look exactly like this.



Squad Builder

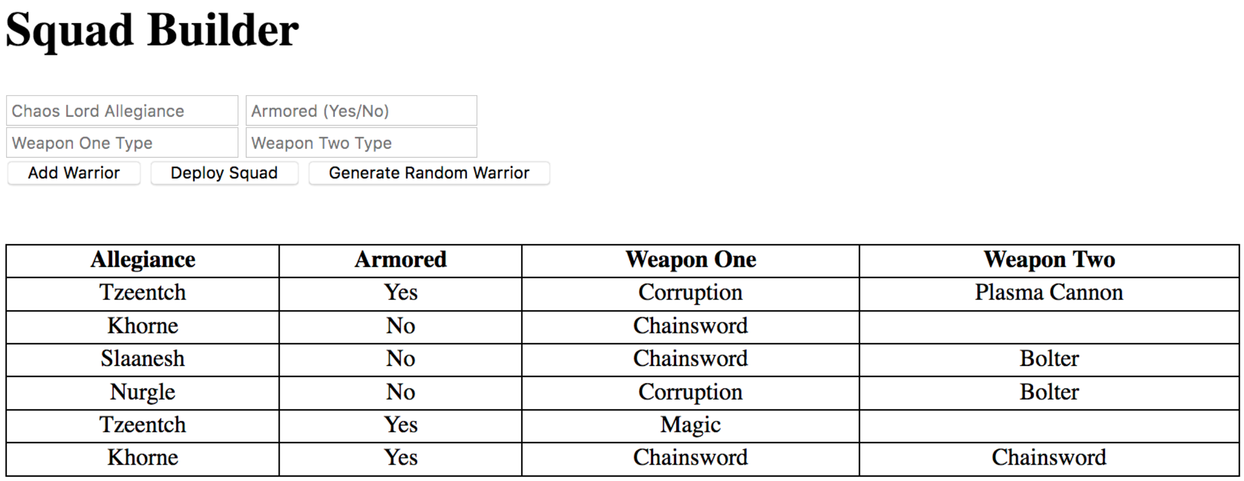
Allegiance	Armored	Weapon One	Weapon Two

3. Create a **javaScript file** named **objects-lab.js**

4. When the "Add Warrior" button is clicked, it should read the data stored in the input text boxes and create a Warrior object from that input. Once it has created the object, it should display a message that a warrior has been created. You need to create warrior objects from the input provided each time Add Warrior is clicked. You will need to store the warriors you create.

5. Warrior objects consist of the following properties:
   - allegiance (string)
   - armored (Boolean)
   - weapons (array of strings)
   - isArmored (function that checks the value of armored and returns "Yes" or "No" based on if armored is true or false respectively.

6. When "Deploy Squad" is clicked, it should populate the table you created with each warrior's information that you have previously stored. **Note**: the output of the Armored column should be Yes or No, not true or false.



Squad Builder

Allegiance	Armored	Weapon One	Weapon Two
Tzeentch	Yes	Corruption	Plasma Cannon
Khorne	No	Chainsword	
Slaanesh	No	Chainsword	Bolter
Nurgle	No	Corruption	Bolter
Tzeentch	Yes	Magic	
Khorne	Yes	Chainsword	Chainsword

7. Add some CSS to your page to at least make the table look decent.

Part 2 (Required)

1. You should create a function to implement the following requirements for user input:

    - A warrior's allegiance must be to Khorne, Slaanesh, Nurgle, or Tzeentch
    - The valid weapons a warrior may carry are
        - Chainsword
        - Plasma Cannon
        - Bolt Rifle
        - Magic
        - Corruption
    - A warrior must have at least 1 weapon.
    - Armored can Yes or No
    - Only warriors with an allegiance of Tzeentch can use Magic.
    - Warriors with an allegiance of Khorne can only carry Chainswords.
    - Warriors with an allegiance to Slaanesh cannot wear armor.
    - Warriors with an allegiance to Nurgle must use Corruption for at least 1 weapon.

    Name this function checkWarriorConstraints. This function should check each constraint/requirement. If the input violates one or more of the constraints, then a meaningful message should be displayed to let the user know. If the requirements are met, then the warrior object should be created and stored somewhere until time for deployment.

2. When the "Deploy Squad" button is clicked, it should only work if **at least 3** warriors have been created. Otherwise it should alert the user they need to create additional warriors.

3. When you are finished, zip all your files together and name it **sfausername-objects-lab.zip** and submit on D2L.

Part 3 (Bonus)

1. Add the Generate Random Warrior button to the HTML
2. When clicked, the Generate Random Warrior should generate a **valid**, random warrior. Valid is based on the constraints described in part 2.
3. The properties of the generated warrior (allegiance, armored, weapons) should be placed in the input text boxes. If the user likes this Warrior, they can then click Add Warrior to add the Warrior to the squad.