

HN VIDEO GAMES

TODAY'S SESSIONS

#1 (09.00 – 12:15): ADVANCED PROGRAMMING

#2 (13.00 – 16.15): GAME DESIGN THEORY

#3 (16.30 – 17.30): TPA - PREVENT

RESOURCES



GitHub

Session Resources are available via
your REBOOT GAMES GitHub Account

- Reboot Games / UMLBasics

ADVANCED PROGRAMMING

Session Objectives: Learning Objective 2 - Design a Series of UML Class Diagrams

- P2 Design and build class diagrams using a UML tool.
- M2 Define class diagrams for specific design patterns using a UML tool.
- D2 Define/refine class diagrams derived from a given code scenario using a UML tool

Today's Session

1. Introduction to UML
2. UML and Object Oriented (OO) Design Concepts
3. UML and OO Design Analysis
4. UML Building Blocks
5. UML Notations
6. Session Task; Individual practical activity building on your work from last week

INTRODUCTION TO UML

UML (Unified Modelling Language) is a standard language for specifying, visualizing, constructing, and documenting the artefacts of software systems. UML was created and first specified (UML 1.0) in January 1997 by the Object Management Group (OMG); Initially it was designed to capture the behaviour of complex software and non-software system but has now become an standard in software engineering and programming.

Wider Reading

There is some excellent material online with regards to UML, search:

“UML IBM CLASS DIAGRAM”

“UML CLASS EXAMPLES”

In addition, a full UML Reference document is available via your GitHub Repo

NOT A PROGRAMMING LANGUAGE

- UML stands for Unified Modelling Language.
- UML is different from common programming languages such as C++, Java, C#, etc.
- UML is a pictorial language designed to create blueprints for software.
- UML is a general purpose visual modelling language to visualize, specify, construct, and document software systems.

Although UML is not a programming language its tools can convert UML diagrams into code; UML has a direct relationship with object oriented analysis and design.

A PICTURE IS WORTH A THOUSAND WORDS

There are a number of goals when using UML but the most important is to define some general purpose representation of a system that is easy to create, understand and use.

UML creates a conceptual model of a system.

- A conceptual model can be defined as a model which is made of concepts and their relationships.
- A conceptual model is the first step before drawing a UML diagram. It helps to understand the entities in the real world and how they interact with each other.

UML describes Object-Orientated real-time systems.

It is very important to consider the conceptual model before proceeding gradually.

OBJECT ORIENTATED CONCEPTS

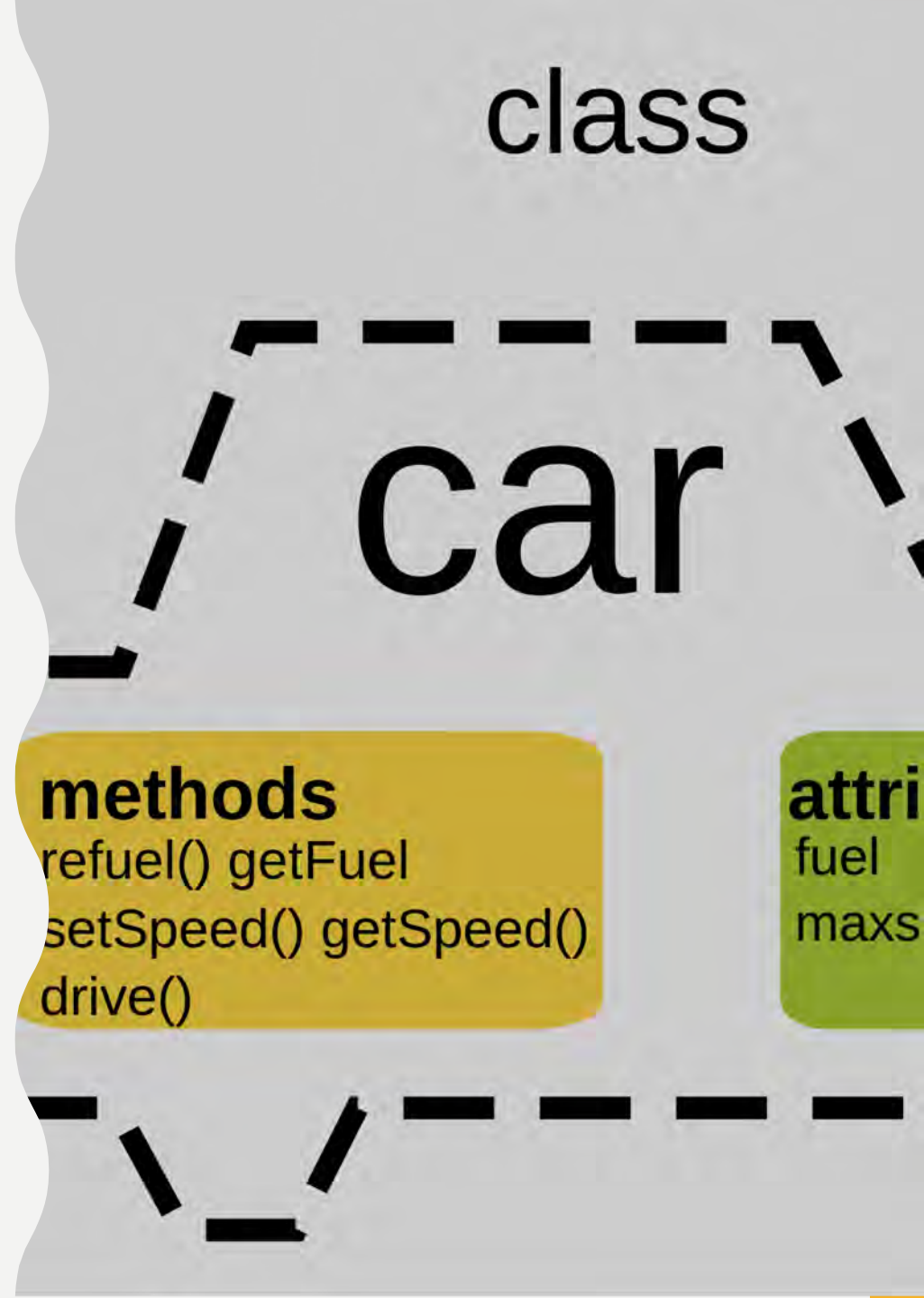
An OBJECT contains both DATA and METHODS that control the data. The data represents the state of the object.

A CLASS describes an object (a blueprint), classes can also be used to form hierarchies to model real-world system.

A HIERARCHY is represented as inheritance between classes, this inheritance be associated in different ways (as per the system's requirements).

OBJECTS are the real-world or virtual entities that can be created and used. Objects also incorporate concepts such as:

- Abstraction
- Encapsulation
- Inheritance
- Polymorphism



ABSTRACTION – DEFINITION

The behaviour of an real world
or virtual entity

Different viewpoints can have
different levels of abstraction; you can
use this to hide various levels of
complexity so that the reader only
needs to focus on the important
detail for that “view”



ENCAPSULATION – DEFINITION

The mechanism of binding data together and hiding it from the “outside world”

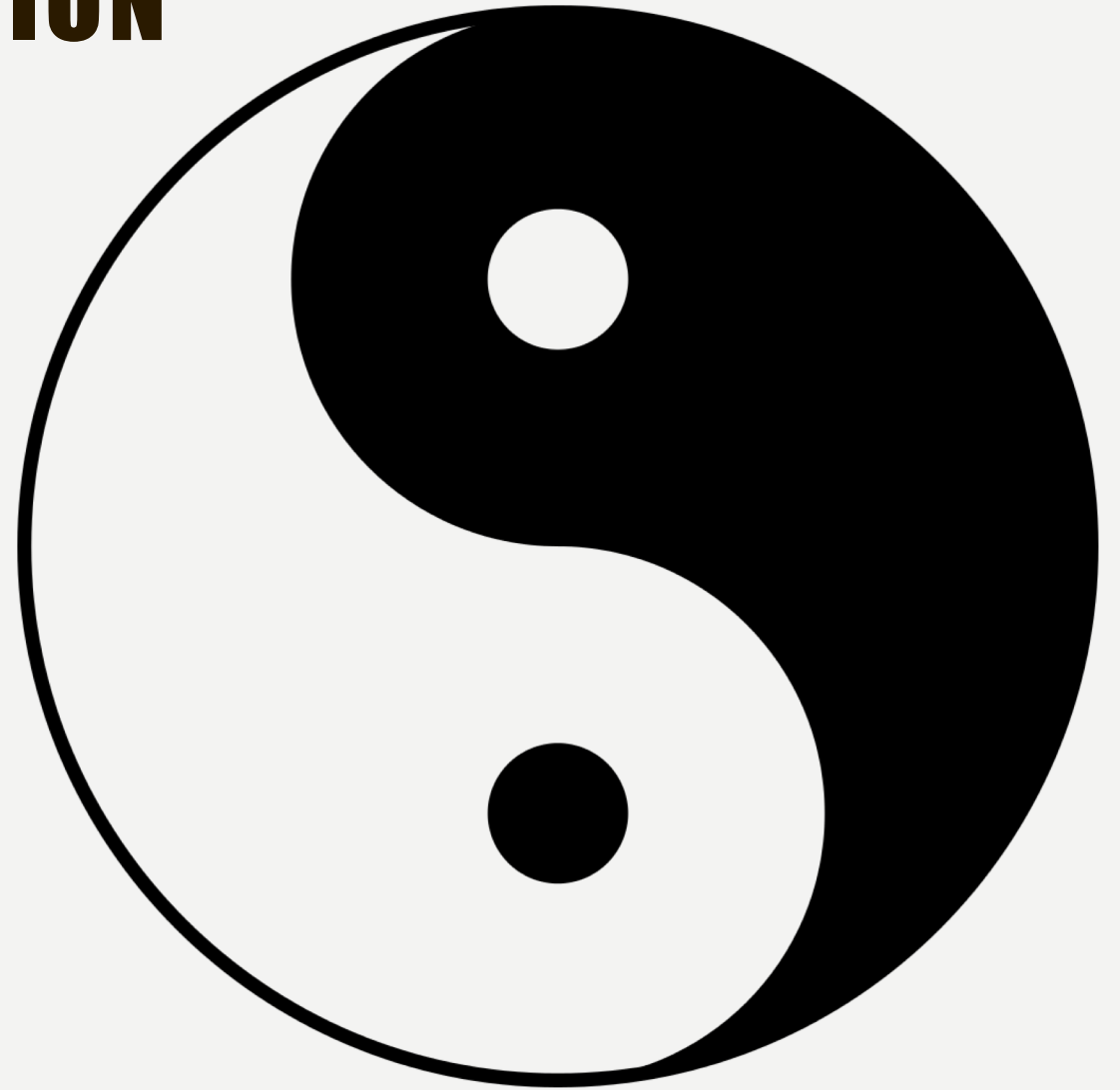
In other words, the term refers to the process of an object controlling how “outside” mechanisms access and use its internal data; a bit like an API



INHERITANCE – DEFINITION

The mechanism of making new classes (blueprints) from and/or by combining existing ones.

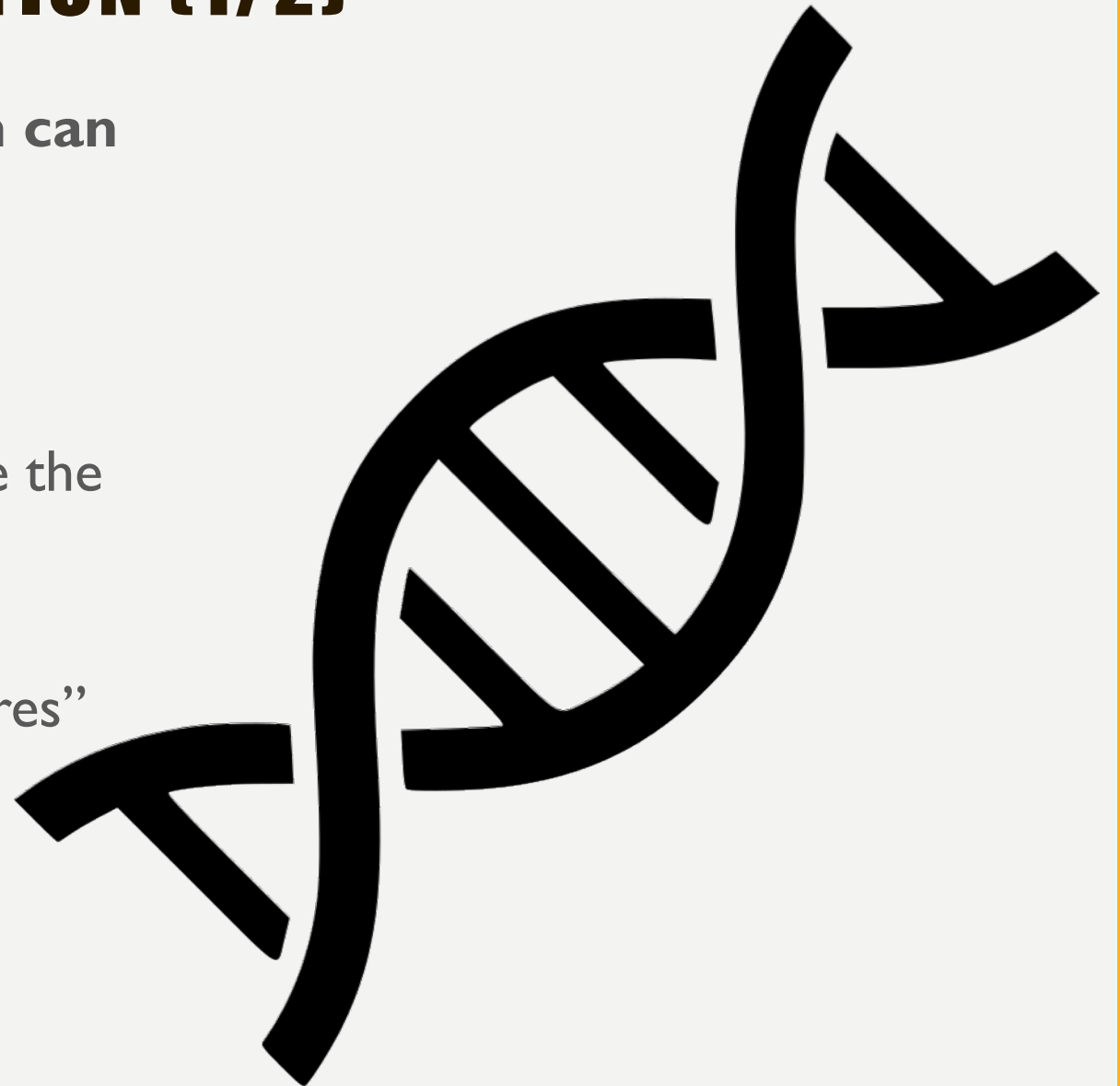
In essence, this is the process of one specific class (called a derived or sub class) incorporating the structure and behaviour of a more general class (called superclass, base class or parent).



POLYMORPHISM – DEFINITION (1/2)

Describes the object's mechanism can exists in different forms.

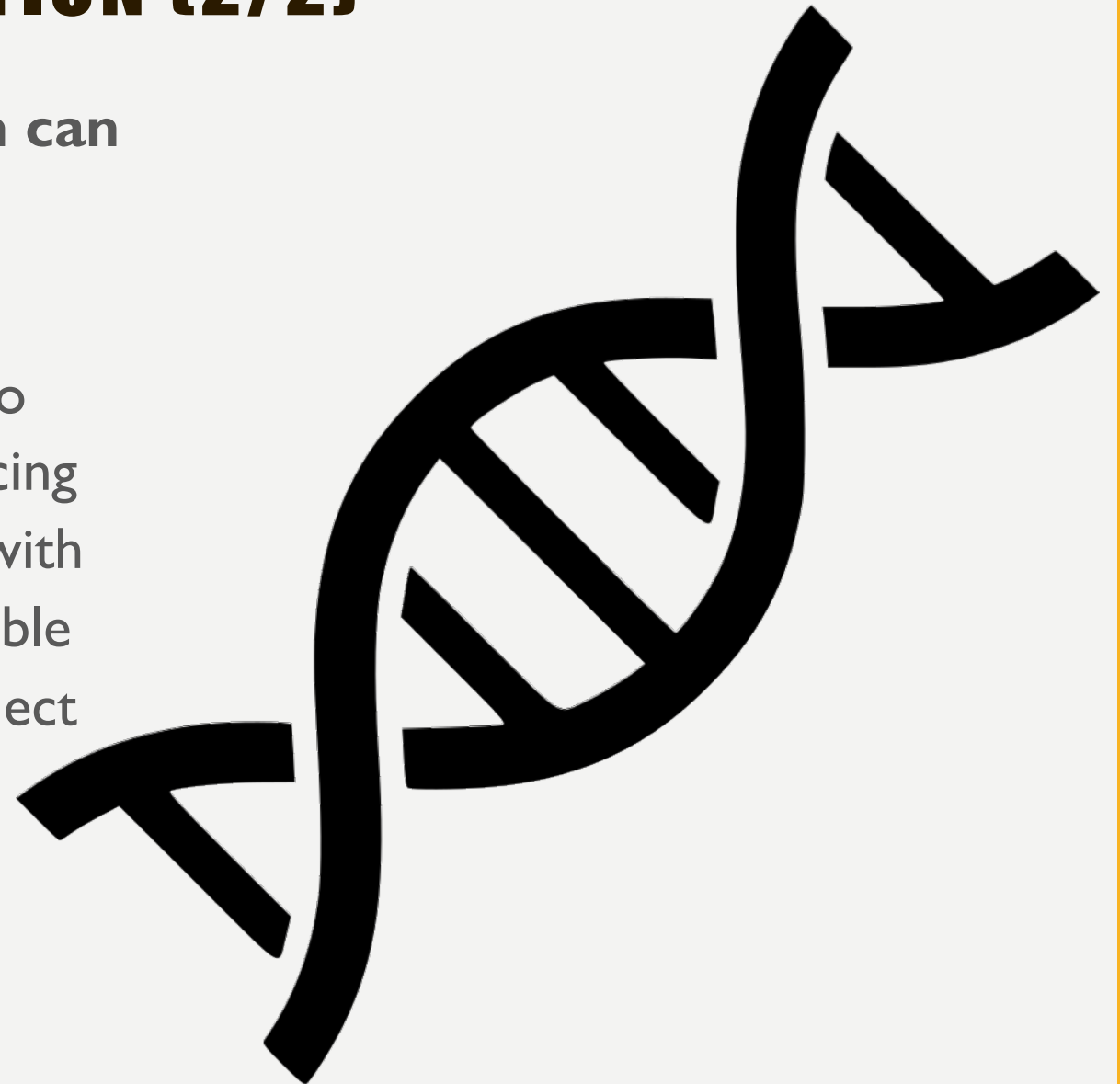
In essence, we can implement multiple methods within the same class that use the same name but with a different set of parameters to change the mechanisms behaviour (by setting different “signatures” we can “overload” the method).



POLYMORPHISM – DEFINITION (2/2)

Describes the object's mechanism can exists in different forms.

In addition to “overloading” we can also ”override”. Overriding refers to replacing one inherited class within a hierarchy with a completely different one thus being able to alter the entire behaviour of the object being used.



OO ANALYSIS AND DESIGN

Object Oriented analysis and design can be defined (simply) as an investigation in to the objects of a system and their collaboration.

The most important purpose of an efficient OO analysis is to identify the objects of a system that need to be designed. Once the objects, their relationships are fully identified and understood can a design is produced.

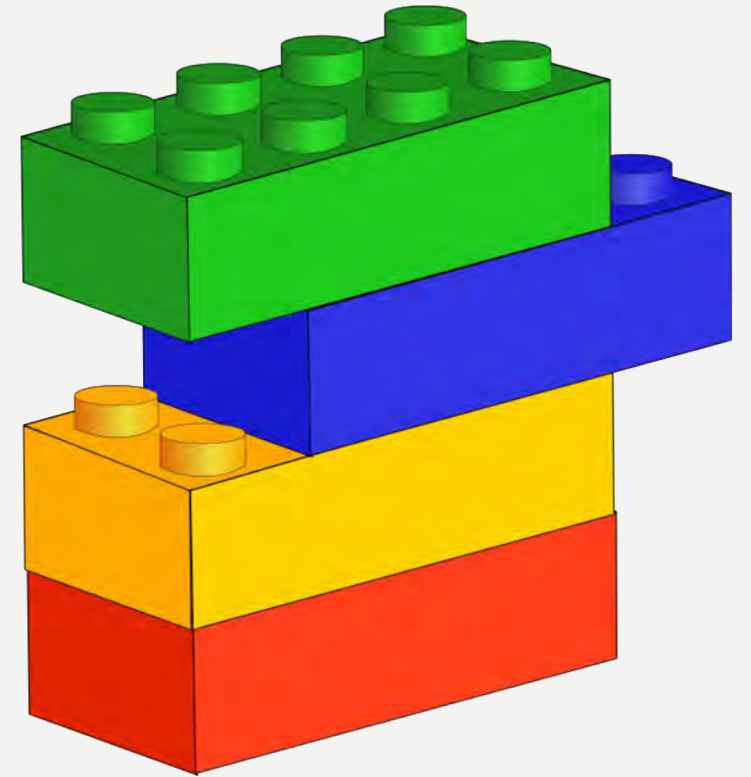
OO Analysis and Design:

1. Identifying (and describe) the objects of a system in a full and proper way; include responsibilities (the functions of the object)
2. Identifying their relationships; consider how you can fulfil the requirements of the objects
3. Make a design that can be converted to an executable using an OO language (C++)

UML BUILDING BLOCKS

The building blocks of UML are:

- Things
- Relationships
- Diagrams



THINGS

Probably the most important building blocks of UML. Things can be —

- **Structural things:** defines the static parts of the model, represent the physical and conceptual elements (e.g. class, Interface (operations/responsibilities), collaboration (interaction between elements), use-case (actions performed for a specific goal), components (parts of the system) and node (a element that exists at runtime), etc)
- **Behavioural things:** consists of the dynamic parts of UML models (e.g. interactions (message exchanges), state machines (sequence of stages an object goes through in response to events), etc.)
- **Grouping things:** the mechanism to group UML elements of a model together (e.g. package, etc)
- **Annotational things:** a mechanism to capture comments, descriptions and remarks of UML element

RELATIONSHIPS

Relationships is another most important building block of UML. It illustrates how elements are associated with each other and thereby describes the the functionality of a system. There are 4 types of relationships:

1. **Dependency** - a relationship between two things in which a change to one affects the other.
2. **Association**- a set of links that illustrate the connections within the elements of a UML model; it also highlights the number of objects within the relationship.
3. **Generalisation** – a relationship which connects a specialised element with a generalised one; basically this shows inheritance.
4. **Realisation** - a relationship in which two elements are connected but one describes some responsibility that is not implemented but used by the other; typically exists in case of interfaces

DIAGRAMS

Diagrams are the final output of the UML process. All the documented elements and relationships are used to make a complete UML diagram of a system.

There are nine principle diagrams:

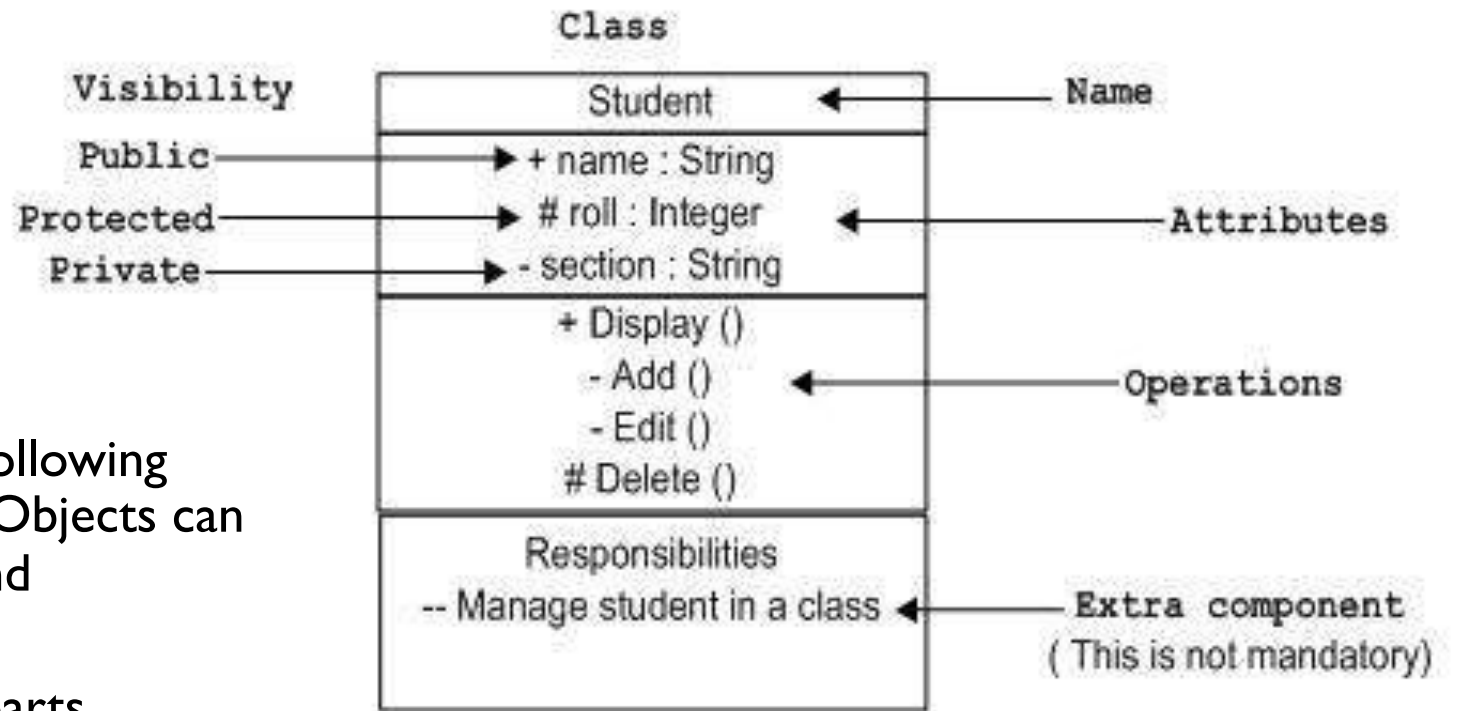
1. Class diagram
2. Object diagram
3. Use case diagram
4. Sequence diagram
5. Collaboration diagram
6. Activity diagram
7. Statechart diagram
8. Deployment diagram
9. Component diagram

CLASS NOTATIONS

UML *class* is represented by the following figure. Classes represent objects, Objects can be anything that has properties and responsibility.

The diagram is divided into four parts.


1. The top section is used to name the class.
2. The second one is used to show the attributes of the class.
3. The third section is used to describe the operations performed by the class.
4. The fourth section is optional to show any additional components.



OBJECT NOTATIONS

Objects are represented in the same way as Class; the only difference being the name is underlined as shown in the following figure.

<u>Student</u>
+ name : String # roll : Integer - section : String
+ Display () - Add () - Edit () # Delete ()



This TASK will require you to do some Research on CLASS diagrams, search online using “UML CLASS EXAMPLES” as a good starting point

Also, when you have a spare minute, check out:

<https://www.ibm.com/developerworks/rational/library/content/RationalEdge/sep04/bell/index.html>

TASK

DESIGN A SIMPLE ROLE PLAYING GAME

INDIVIDUAL SESSION TASK

Build and **CRITICALLY** reflect on the work you did last session (Game Mechanics) to help you complete the following:

Design and Document Suitable CLASS diagrams for the following real-world, GAME objects –

- PLAYER CHARACTER
- END OF LEVEL BOSS
- MOB ENEMY
- GEOGRAPHIC/MAP LOCATION
- GAME OBJECTS
- WEAPONS (RANGE/MELEE/MAGIC)

Try to include abstraction & inheritance; don't forget about attributes, responsibilities, functions, methods (consider game play and your experience)



GAME DESIGN THEORY

SESSION 2

GAME DESIGN THEORY

Session Objectives: Learning Objective 4 (partial) - Design a high concept pitch for your original game

- P6 Create an original game concept.
- P7 Develop an original Game Design High Concept presentation.
- M4 Conduct peer-reviews using your original game concept and document any feedback given

Today's Session

1. Discuss High Concept Game Pitches
2. Review BioShock Pitch Document
3. Session Task

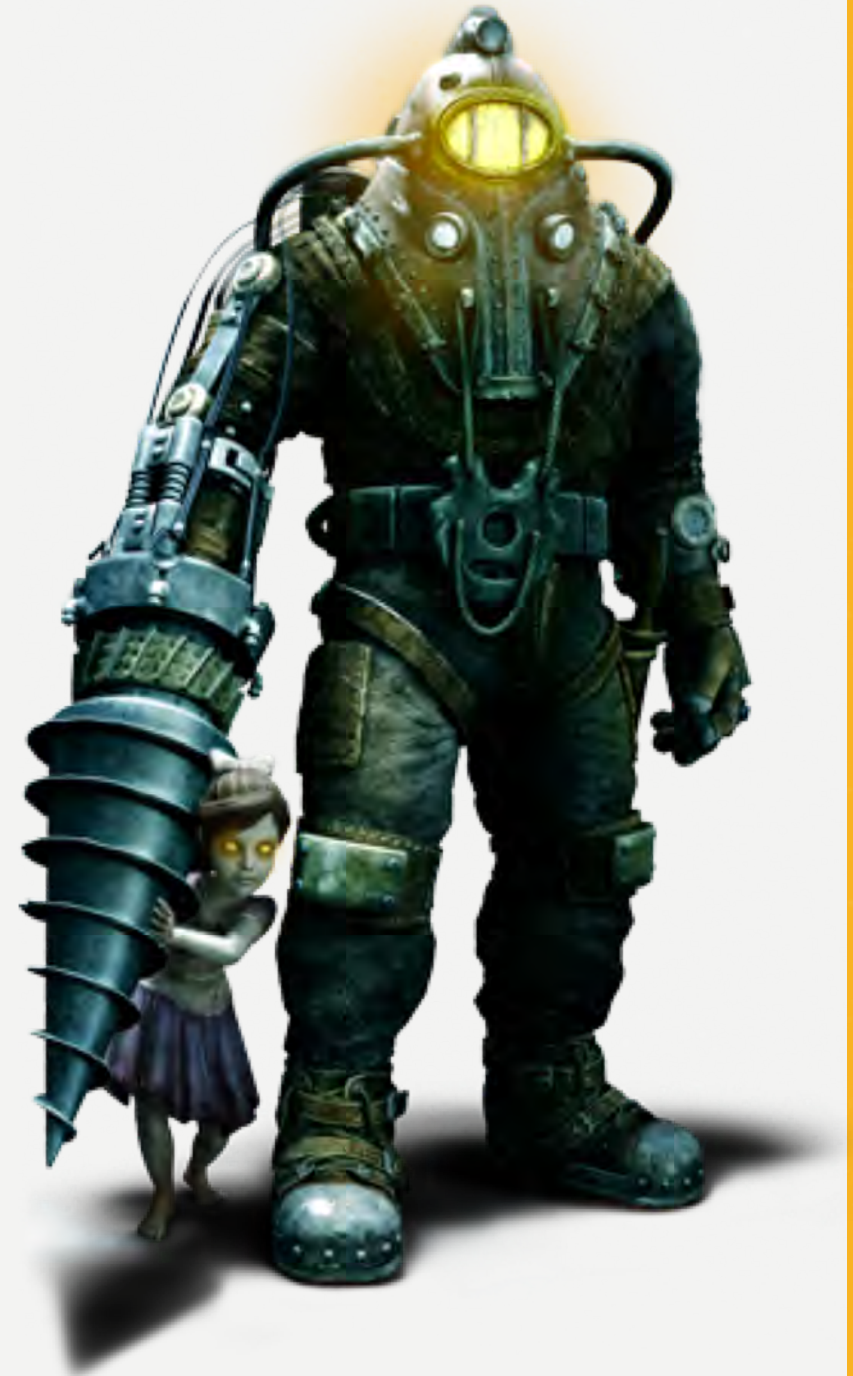
HIGH CONCEPT GAME PITCHES - PURPOSE

The Pitch should be a ten to fifteen page document presented in a horizontal orientation; it can be in any format (PowerPoint, PDF, etc). The purpose of the pitch is to showcase what your game is all about, what are its unique selling points, what is the value of the game for the publisher and can the development team actually deliver the production.



HIGH CONCEPT GAME PITCHES - THEME

Your Pitch should showcase the visuals and the mood of the game in a proper fashion. In the best case scenario, every page should have images related to the game or at least the theme of the pitch should match with the mood of the game. The pitch needs to represent the game as a finalised and mature product. This means that the art work should be of the highest quality possible and be specifically made for the pitch; of course, drafts and concept images are good to use but they also should be of high quality!



HIGH CONCEPT GAME PITCHES - CONTENT

- **Cover** - catch the attention of the reader at the first glance
- **Overview** - key information of the game on a single page. A one sentence pitch fits well at the top of the page continued with short bullet points of the game's main features, hooks and unique selling points; perhaps consider "game A meets game B in a fantasy environment"
- **Gameplay** - what the game is going to look like when it will be finished, preferably have some action going on or something that the reader would find interesting
- **Metagame** - Metagame means the game outside the main game, e.g., upgrading your character, selecting weapons or managing your inventory, etc.
- **Content & Features** - give more information about main gameplay and metagame, special features, storylines, challenges, etc
- **Market and Competitors** – give insight into the value of the game by showing the potential it can have in the current market and its value proposition



The background is a dark, atmospheric image of the BioShock title screen. It features the word "BIO" in large, stylized, metallic letters at the top, and "SHOCK" in similar letters below it. The letters are weathered and have a mechanical, industrial feel. Below the title, there is smaller text that reads "FIRST PERSON ACTION HORROR" and "FROM THE CREATORS OF SYSTEM SHOCK 2". The overall color palette is dark with shades of grey, black, and some muted green and yellow highlights on the letters.

REVIEW BIO SHOCK PITCH

Using this opportunity to plan the sort of information and approach you will need for your pitch

GROUP SESSION TASK

Building on the work you did in your group last session work to complete the following:

Design, Create and Review a HIGH CONCEPT pitch for your group game. Your pitch must be of a suitable quality.

Use the aspects discussed today to help plan and prepare your work.

Remember:

- Cover
- Overview
- Gameplay
- Metagame
- Content & Features
- Market and Competitors





PREVENT

SESSION 3 - TPA

TPA - PREVENT

Session Objectives

- Short Presentation Highlighting and Discussing what PREVENT is
- ITSLearning Activity

PREVENT

2018/2019

SIDE BY SIDE

EDUCATION & TRAINING
FOUNDATION



OBJECTIVES

Identify what the PREVENT agenda covers.

Demonstrate an understanding of PREVENT

Recognise key terminology linked to PREVENT

Q&A

What is PREVENT?

Any ideas?

WHAT IS PREVENT?

The aim of the Prevent duty is to identify people who may be vulnerable to radicalisation.

As a college, we aim to raise awareness of the signs of radicalisation and support students at risk of radicalisation.

This session will help you know what to do if you think you or someone you know may be at risk of radicalisation.

WHY DO WE LEARN ABOUT PREVENT?

- To **PREVENT** young people from being involved in criminal activity.
- We have a duty to **PREVENT** young people from being radicalised and to make sure you know what to do if someone you know is being radicalised.
- We must **raise awareness** of the types of terrorism that surround us. We will **support** anyone who is at risk through the appropriate channels.

TASK: COMPLETE THE FOLLOWING MATCHING ACTIVITY 0

DISCUSS WHAT DEFINITION BEST DESCRIBES THE TERMINOLOGY.

Terminology:		Definition:
Ideology		A person who uses unlawful violence and intimidation, especially against civilians, in the pursuit of political aims
Radicalisation		A system of ideas and ideals, especially one which forms the basis of economic or political theory
Terrorism		The process of protecting vulnerable people, from crime, abuse or terrorism- related activity
Safeguarding		The process by which a person comes to support terrorism and extremist ideologies associated with terrorist groups
Terrorist		The unlawful use of violence and intimidation, especially against civilians, in the pursuit of political aims

Terminology:		Definition:
Ideology		A person who uses unlawful violence and intimidation, especially against civilians, in the pursuit of political aims
Radicalisation		A system of ideas and ideals, especially one which forms the basis of economic or political theory
Terrorism		The process of protecting vulnerable people, from crime, abuse or terrorism- related activity
Safeguarding		The process by which a person comes to support terrorism and extremist ideologies associated with terrorist groups
Terrorist		The unlawful use of violence and intimidation, especially against civilians, in the pursuit of political aims

RECAP:

- What is the main aim of PREVENT?
- Why do we cover PREVENT at West Herts college?
- Identify key terminology and what it means
- Where would you go for further advice and support?
- **What's Next:**
- Log onto the following website:
- <http://www.etflearners.org.uk/>
- Create an account and complete the 'Side by Side' radicalisation unit. You will watch a few short clips and complete a short multiple choice quiz.