Option D – Object oriented programming (OOP)

13. (a) *Award **[1 max].***
These are data types that are **pre**-defined / fundamental / basic in the programming language;
A data type that is always assigned a value (in the memory);
They are the building blocks of the composite data types / classes / objects;
The types that are implemented directly as bit patterns (by a Java compiler);

(b) (i) *Award **[1 max]** for any suitable exampl**e.** Allow a description.*
For example
`hasParking;`
`hasPool;`
`hasGarden;`
Whether the property has been sold or not;

(ii) *Award **[1 max]** for any suitable exampl**e.** Allow a description.*

For example
`numberRooms;`
`area;`
The number of bathrooms;

(c) (i) *Award **[1 max].***
Aggregation (allow Property 'has a' Owner);

(ii) *Award **[1 max].***
Inheritance (allow House 'is a' Property);
House is a subclass of Property;
House inherits Property;

(d) *Award **[2 max].***
*Award **[1]** for distinguishing between a class and an instantiation at the definition level*;
*Award **[1]** for including an example*;

A class (e.g. House class) is a blueprint / definition / specification (that defines all variables and methods that are needed) and an instantiation of a class is creates a (new) object of that class ***[1]*** (e.g. *a (new) object of an actual house* giving its actual address etc ***[1]***;

(e) *Award **[2 max].***
A static variable is used when it is to be a class variable (belongs to the class not to the instantiation);
It is used when all objects of that class are to have the same value for the static variable;
It will not be instantiated when a new object is created;
If the value of a static variable is changed, it will be changed in all instances of that class;
It does not require an object of that class to be accessed;

(f) *Award **[2 max].***
Both classes House and Apartment need a static (integer) variable `count`;
that is incremented whenever a new object of that class is created;

A static variable (e.g `total`) could be defined in the Property class;
Which is incremented every time a House or Apartment object is instantiated;

A static method in the Property class retrieves the total number of house and apartments;
From static total variables that are in each of the 2 sub-classes;

(g) *Award **[3 max]**.*
*Award **[1]** for the identifier (left-hand side);*
*Award **[1]** for use of `new`;*
*Award **[1]** for `House[10];`*

```
House[] wishList = new House[10];
```