1a) Four legal identifier names are

- *myVariable*
- *_score*
- *$total*
- *number_of_items_2*

1b) Four illegal identifiers and why

- *my variable*
  - Illegal because identifiers can't have spaces.
- *2ndPlace*
  - Illegal because identifiers can't begin with a number.
- *Class*
  - Illegal because it's a reserved keyword in Java.
- *my-variable*
  - Illegal because identifiers can't have a hyphen.

2a)
```
int numBeads;
numBeads = 5;
```
2b)
```
int numBeads = 5;
```

3a) The final value of *yourNumber* is 13
3b) The final value of *yourNumber* is 11

4a) *int*
4b) *float* or *double*
4c) *int*
4d) *float* or *double*
4e) *boolean*
4f) *char*

5a) A primitive data type is a type of data that holds just a single item, and it is otherwise referred to as a built-in data type. Primitive data types available in Java include: *int*, *double*, *char* and *boolean*. An abstract data type, or class, is a data type that is able to hold both data and methods.
5b) A class is a blueprint, or template, for objects. An object is a variable declared with a class. In this way, a class outlines the structure along with attributes and behaviours, while an object is a real, tangible version of a class.

11a) `y = (int) (j * k);`

- With Rounding:
  `y = (int)Math.round(j * k);`

11b) No type casting is needed because both values are *doubles*, so the result is a *double*.
11c) No type casting is needed since everything is *double*.
11d) *j = (int)k;*

- With Rounding:
  j = (int)Math.round(k);

11e) No type casting because you can automatically assign an *int* to a *double*.
11f) All types match; no casting is needed.

https://g.co/gemini/share/f88dea36f217