# Solutions 2.1

Learn C With Babbo

1. a) Declared
   To say `my_var` was initialized, we would've had to assign it on declaration, as in

   ```
   int my_var = 10;
   ```

   Likewise, assignment requires an equals sign. We haven't covered pineapples yet.

2. c) Data Type

3. c) Variable, Identifier.

4. (a) Not initialized, declared

   (b) Initialized

   (c) Not initialized, declared and assigned

   (d) Not initialized, assigned

5. b) Literal

6. `my_var2` is equal to 2. Assignment of `my_var1` to 4 does not change the value of `my_var2`: the code

   ```
   int my_var2 = my_var1;
   ```

   Assigns `my_var2` to the value of `my_var1` **at the time of assignment**, it does not continue to "update" to new values of `my_var1`, unless you assign `my_var2` to `my_var1` again.

7. a) Valid Identifier
   b) Valid Identifier
   c) Invalid Identifier: identifiers may not begin with a digit
   d) Valid Identifier
   e) Valid Identifier
   f) Invalid Identifier: identifier may only contain alphanumeric characters(a-z, A-Z, 0-9) and underscores, not symbols like !

g) Invalid Identifier, for the reasons listed above.
h) Valid Identifier

8. 
```
printf("%d\n", my_var);
```