

1. Line 20 in ATM_Interface. I used print() instead of println() because the user was required to input directly after the string "Amount (in \$) to withdraw". If println() was used, it would make the user input in a newline after the string.
2. The \ character is used as an escape sequence, meaning that depending on the character that follows it, it performs a different action. Some characters that normally print (e.g. \n, \t) would perform its special action, but characters that normally perform a specific action would instead print raw (e.g. \", \\\). Line 17 in ATM_Interface used the escape sequence because it needed to print the raw backslash character. So \\ was used to print a backslash \.
3.
 - a) int withdrawal in ATM_Interface. The int data type is the best choice as the withdrawal amount the user is prompted for would be an integer (because it is in dollars), and decimal points would not be necessary.

double height in FenceEstimate. The double data type is the best choice as the height amount the user is prompted for could include decimals (e.g. 2.5 meters).
 - b) The char data type holds a character value such as a letter or an ascii number that represents a letter (e.g. a, b, c, 1, 2, 3). This was not needed as none of the programs needed to hold a character value.

The string data type holds a string, which are multiple characters that are "stringed" together (e.g. "Hello, World!", "Hello Mr. Skuja :D"). This was not needed as none of the programs needed to hold a string value.

4. Constants are variables that will always remain the same while the program is running (hence the name “constant”). Using constants is good coding practice whenever a value will always be constant, such as tax (13%) or a price of an item (e.g. \$1.75). FenceEstimate uses constants to represent the price of wired and wooden fences. The costs were stored in separate constant double values and would be multiplied by how many meters of fencing were ordered. I chose to use these constants because they would always remain the same, and instead of typing the actual price whenever I needed it, I could instead type the variable holding the constant value.