|  |  |
| --- | --- |
| Full Name | Laure Patera |
| Student ID | 1925466 |

Initial Design

1. Output to the user, “You are walking into a pet store with your friend and considering buying a new pet together. Your friend is buying the pet, but you will be buying the supplies with a budget of $50. On the side closest to you there are walls of fish, but towards the opposite side of the store are aisles with cages of various rodents and birds.
2. Create the variable budget as a float which is equal to 50
3. Create the variable pet\_type and ask the user to input a string based on the question: “Of the options your friend is deciding between, would you rather get a betta, a guinea pig, or come back another day?”
4. If the variable pet\_type equals “guinea pig” and “betta” output to the user: “Your friend’s budget is not high enough for that”
5. If the variable pet\_type equals “betta” create the variable name and ask the user to input a string that will be stored with it, using the question, “Aww what are you naming it?”
   1. Output to the user, “(name) is the perfect name!”
   2. Output to the user, “Now you have to decide between food. These are the options on the shelf in front of you.”
   3. Output on a new line: “Food 1: Costs $10.25 and (name) is already used to eating it.”
   4. Create the variable food1 as a float which is equal to 10.25
   5. Output on a new line: “Food 2: Costs $3.99 and is the cheapest option if you want to save.”
   6. Create the variable food2 as a float which is equal to 3.99
   7. Output on a new line: “Food 3: Costs $15.75 and has a three-month supply, which is much more than either other option.”
   8. Create the variable food3 as a float which is equal to 15.75
   9. Ask the user to input either “Food 1”, “Food 2” or “Food 3” with the question, “Which food would you like to buy for (name)?”
   10. Create the variable food and set it equal to the value of whichever option is inputted
   11. Calculate the remaining budget using budget – food, then output to the user, “You have $ (budget amount) left that you can spend.”
6. If the variable pet\_type equals “guinea pig” create the variable guinea\_pig\_number and ask the user to input an integer that will be stored with it, outputting: “Nice! How many guinea pigs do you want to get?”
   1. If guinea\_pig\_number is greater than or equal to 4, output to the user “Hmm. If you think you can handle that.”
   2. Otherwise if guinea\_pig\_number is less than or equal to 1, output to the user “Are you sure it won’t be lonely?”
   3. Otherwise output to the user “That sounds like a good number!”
7. Output to the user: “Now you have to decide between food. These are the options on the shelf in front of you.”
   1. Output on a new line: “Food 1: Costs $12.85 and (name) is already used to eating it.”
   2. Create the variable food1 as a float which is equal to 12.85
   3. Output on a new line: “Food 2: Costs $5.99 and is the cheapest option if you want to save.”
   4. Create the variable food2 as a float which is equal to 5.99
   5. Output on a new line: “Food 3: Costs $20.75 and has a three-month supply, which is much more than either other option.”
   6. Create the variable food3 as a float which is equal to 20.75
   7. Ask the user to input either “Food 1”, “Food 2” or “Food 3” with the question, “Which food would you like to buy for (name)?”
   8. Create the variable food and set it equal to the value of whichever option is inputted
   9. Calculate the remaining budget using budget – food, then output to the user, “You have $(budget amount) left that you can spend.”
8. If the variable pet\_type does not equal “betta”, “betta fish”, or “guinea pig” output: “You may not have a pet, but you saved money and avoided a potentially hasty decision.”