

Use Of Selection:

Use of selection can be seen throughout the code for example:

```
if (customer == "") //check if the user inputed a customer name
{
    MessageBox.Show("Please Enter a Customer Name."); //Prompt the user if they didnt
}
else
{
    if (!inputValid || pizza > 2 || pizza < 1) //if they did enter a name then check
that the number of pizza are valid
    {
        MessageBox.Show("Please Enter a Valid Number of Pizzas. (1 or 2)"); //if
not prompt the user to enter a valid amount
    }
    else
    { //if num pizzas is valid determine if to show inputs for 1 pizza or 2 pizzas
        if (pizza == 1) //if there is 1 pizza selected
        {
            visible_labels(pizza1Labels, true); //show the labels for pizza 1
            visible_labels(pizza2Labels, false); //hide the labels for pizza w
            visible_checkboxes(pizza2CheckBoxes, false); //hide the checkboxes for
pizza 2
            visible_checkboxes(pizza1CheckBoxes, true); //show the checkboxes for
pizza 1
            pizza1ToppingIn.Visible = true; //show the topping input for pizza1
            pizza2ToppingIn.Visible = false; // hide the topping input for pizza 2
        }
        else //if there are 2 pizzas
        {
            visible_labels(pizza1Labels, true); //show all the inputs
            visible_labels(pizza2Labels, true);
            visible_checkboxes(pizza2CheckBoxes, true);
            visible_checkboxes(pizza1CheckBoxes, true);
            pizza1ToppingIn.Visible = true;
            pizza2ToppingIn.Visible = true;
        }
    }
}
}
```

In this segment, if statements and conditions are used to check if the User input a valid Customer Name and number of pizzas, if they did then they see the inputs for the number of pizzas they are ordering, if they didn't, they see a prompt asking them to provide valid inputs.

Use Of Iteration:

Use of Iteration in the code can be seen in the function, **visible_labels**:

```
//function iterates through the list of labels and sets the visible property to the bool parameter vis
```

```
private void visible_labels(Label[] labels, bool vis)
{
    for (int i = 0; i < labels.Length; i++) //iterates through the labels array
    {
        labels[i].Visible = vis; //sets the label at the current index as visible
        or invisible based on vis
    }
}
```

This function iterates through the array of labels passed as a parameter for the function, then setting the label at the value of the current index of the loop to visible or invisible based on the value (true or false) of the parameter vis.