# Data Structures Used

## Vector: Found in line city.cpp - line 84

```cpp
std::vector<Food> City::getNonZeroFoods()
{
        std::vector<Food> nonZeroFoods;
        for (auto it = foods.begin(); it < foods.end(); ++it)
        {
                if (it->getQuantity() > 0)
                {
                        nonZeroFoods.push_back(*it);
                }
        }
        return nonZeroFoods;
}
```

## Linked List (Doubly Linked List):

QStringList is equivalent to the std doubly linked list in c++. Found in viewwindow.cpp - line 16

```cpp
void ViewWindow::populateCombo()
{
    QStringList cities;      /// Create QStringList (Doubly Linked List of strings)

    QSqlQuery query;
    query.exec("SELECT name FROM city WHERE active=1");

    /// Populate cities QStringList with cities from city table
    if(!query.exec())
    {
        qDebug("Failed to populate comboBox");
    }
    else
    {
    while(query.next())
        {
            cities.push_back(query.value(0).toString());
        }
    }

    for(int i = 0; i < cities.size(); ++i)
        ui->cityComboBox->addItem(cities.at(i));
}
```

# Array:

One of the trip plans is loaded using an Array of QStrings. Found on line
108 of cityselectwindow.cpp

```cpp
void CitySelectWindow::setParisPlan()
{
    QString parisCities[13] = { "Paris", "Amsterdam", "Berlin", "Brussels", "Budapest", "Hamburg", "Lisbon",
                                "London", "Madrid", "Prague", "Rome", "Stockholm", "Vienna"};

    for (int i = 0; i < (sizeof(parisCities)/sizeof(parisCities[0])); ++i)          ⚠ comparis
    {
        cityNames.push_back(parisCities[i]);
    }

    ui->cityTable->setRowCount(cityNames.size());                    ⚠ implicit conversion changes signedn
    ui->cityTable->setColumnCount(3);
    ui->cityTable->setHorizontalHeaderItem(0, new QTableWidgetItem("Cities to Visit"));
    ui->cityTable->setHorizontalHeaderItem(1, new QTableWidgetItem("11 Cities"));
    ui->cityTable->setHorizontalHeaderItem(2, new QTableWidgetItem("13 Cities"));
    ui->cityTable->setSelectionMode(QAbstractItemView::NoSelection);

    QLabel *name;
    for (int i = 0; i < cityNames.size(); i++)                       ⚠ comparison of integers of different sig
    {
        name = new QLabel();
        name->setText(cityNames[i]);                                ⚠ implicit conversion changes signedn
        ui->cityTable->setCellWidget(i, 0, name);
    }
    ui->cityTable->setCellWidget(0,1, new QRadioButton());
    ui->cityTable->setCellWidget(0,2, new QRadioButton());

}
```