

```

void Controller::uploadFoodsFile()
{
    QString fileName = QFileDialog::getOpenFileName(nullptr, tr("Open File"), "/home/CS1D-Project",
tr("Text Files (*.txt)"));
    QFile file(fileName);

    if (!file.open(QIODevice::ReadOnly | QIODevice::Text))
        qDebug() << "Error reading file.";
    else
    {
        QTextStream in(&file);

        while (!in.atEnd())
        {
            QSqlQuery qry;
            QString city = in.readLine();
            QString food = in.readLine();
            QString cost = in.readLine();

            qry.prepare("INSERT INTO Foods (City, Food, Cost)"
                "VALUES (?, ?, ?);");

            qry.addBindValue(city);
            qry.addBindValue(food);
            qry.addBindValue(cost);

            if (!qry.exec())
                qDebug() << "ERROR READING .TXT ON " << city << ", " << food << ", " << cost;
            else
            {
                qDebug() << "FOOD DATA APPENDED TO .DB: " << city << ", " << food << ", " << cost;

                qry.clear();
            }
        }
    }
}

```

uploadFoodsFile() runs in O(n) time

```

void Controller::createTripList()
{
    QSqlTableModel model;
    model.setTable("Distances");
    model.select();

    for (int i = 0; i < model.rowCount(); i++)
    {
        Trip* entry = new Trip();
        entry->setStartCity(model.record(i).value("StartCity").toString());
        entry->setEndCity(model.record(i).value("EndCity").toString());
        entry->setDistance(model.record(i).value("Distance").toInt());

        this->tripList.append(entry);
    }
}

```

createTripList() runs in $O(n)$ time

```

void Controller::createFoodList()
{
    QSqlTableModel model;
    model.setTable("Foods");
    model.select();

    for (int i = 0; i < completedTripList.size(); i++)
    {
        for (int j = 0; j < model.rowCount(); j++)
        {
            if (model.record(j).value("City").toString() == completedTripList[i]->getStartCity())
            {
                food* entry = new food();
                entry->setCity(model.record(j).value("City").toString());
                entry->setName(model.record(j).value("Food").toString());

                QString costAsString = model.record(j).value("Cost").toString();
                costAsString.remove(0,1);
                double costAsDouble = costAsString.toDouble();

                entry->setCost(costAsDouble);

                this->foodList.append(entry);
            }
        }
    }

    for (int i = 0; i < model.rowCount(); i++)
    {
        if (model.record(i).value("City").toString() == completedTripList.back()->getEndCity())
        {
            food* entry = new food();
            entry->setCity(model.record(i).value("City").toString());
            entry->setName(model.record(i).value("Food").toString());

            QString costAsString = model.record(i).value("Cost").toString();
            costAsString.remove(0,1);
            double costAsDouble = costAsString.toDouble();

            entry->setCost(costAsDouble);

            this->foodList.append(entry);
        }
    }
}

```

$n^2 + n$

n^2

createFoodList() runs in $O(n^2)$ time

```

void Controller::createCustomTripList()
{
    QSqlTableModel model;
    model.setTable("Distances");
    model.select();

    for (int i = 0; i < customTripListCities.size(); i++)
    {
        for (int j = 0; j < model.rowCount(); j++)
        {
            if (model.record(j).value("StartCity").toString() == customTripListCities[i])
            {
                for (int k = 0; k < customTripListCities.size(); k++)
                {
                    if (model.record(j).value("EndCity").toString() == customTripListCities[k])
                    {
                        Trip* entry = new Trip();

                        entry->setStartCity(model.record(j).value("StartCity").toString());
                        entry->setEndCity(model.record(j).value("EndCity").toString());
                        entry->setDistance(model.record(j).value("Distance").toInt());

                        this->tripList.append(entry);
                    }
                }
            }
        }
    }
}

```

createCustomTripList() runs in $O(n^3)$ time

```

void Controller::uploadCitiesFile()
{
    QString fileName = QFileDialog::getOpenFileName(nullptr, tr("Open File"), "/home/CS1D-Project",
tr("Text Files (*.txt)"));
    QFile file(fileName);

    if (!file.open(QIODevice::ReadOnly | QIODevice::Text))
        qDebug() << "Error reading file.";
    else
    {
        QTextStream in(&file);

        while (!in.atEnd())
        {
            QSqlQuery qry;
            QString startCity = in.readLine();
            QString endCity = in.readLine();
            QString distance = in.readLine();

            qry.prepare("INSERT INTO Distances (StartCity, EndCity, Distance)"
                "VALUES (?, ?, ?);");

            qry.addBindValue(startCity);
            qry.addBindValue(endCity);
            qry.addBindValue(distance);

            if (!qry.exec())
                qDebug() << "ERROR READING .TXT ON " << startCity << ", " << endCity << ", " << distance; n
            else
            {
                qDebug() << "CITY DATA APPENDED TO .DB: " << startCity << ", " << endCity << ", " <<
distance;
                qry.clear();

                Trip* entry = new Trip();
                entry->setStartCity(startCity);
                entry->setEndCity(endCity);
                entry->setDistance(distance.toInt());
                this->tripList.append(entry);
            }
        }
    }
}

```

uploadCitiesFile() runs in O(n) time