

## Плюсы:

1. Организация проекта в виде классов
2. Отсутствие глобальных переменных
3. Не используется множественное наследование
4. Использование стандартных структур данных

## Минусы:

1. Строки длиной больше 80 символов

```
QMessageBox *mb = new QMessageBox(QMessageBox::Question, "Выход", "Вы уверены, что хотите выйти?", QMessageBox::Yes | QMessageBox::No, this);
```

2. Закомментированные куски кода

```
/*  
void Model::cookDishes()  
{  
    needToCookMore = true;  
}  
*/
```

3. Наличии очевидные комментарии

```
QString name;    // название  
float price;    // цена  
int quantity, quantityToCook;    // количество (и какой партией готовить)  
int popularity; // популярность  
QList<bool> requiredUtensils;    // требуемые приборы
```

4. Объявление нескольких переменных в одной строке

```
int quantity, quantityToCook;    // количество (и какой партией готовить)
```

5. Объявление нескольких классов в одном файле

```

class Entity : public QObject
{
    Q_OBJECT
protected:
    QPoint coord; // координата
public:
    explicit Entity(QPoint startCoord, QObject *parent = 0);
    virtual void Draw(QPainter *pntr, QRectF geom); // отрисовка
    QPoint getCoord(); // получение координаты
signals:
public slots:
};

// стена
// с ней ничего нельзя сделать, равно как и просто пройти сквозь нее
class Wall : public Entity
{
    Q_OBJECT
protected:
public:
    explicit Wall(QPoint startCoord, QObject *parent = 0);
    virtual void Draw(QPainter *pntr, QRectF geom);
signals:
public slots:
};

```

## 6. Использование emit в то время как есть специальная функция

```

        emit needsRepaint();
        newthread->start();
    }

void Model::onRepaintRequest()
{
    emit needsRepaint();
}

```

## 7. Большая вложенность

```

if (isDesignMode)
{
    switch (placementID)
    {
        case 1:
        {
            bool done = false;
            for (int i = 0; i < entities.size(); i++)
                if (entities.at(i)->getCoord() == clickPoint)
                {
                    Entity *e = entities.at(i);
                    for (int k = 0; k < countersShortcuts.size(); k++)
                    {
                        if (countersShortcuts[k] > i)
                            countersShortcuts[k]--;
                        else if (countersShortcuts[k] == i)
                            countersShortcuts[k] = -1;
                    }
                    delete e;
                    entities.removeAt(i);
                    freeCells[clickPoint.y() * roomSize.first + clickPoint.x()] = true;
                    done = true;
                    break;
                }
            if (done) break;
            for (int i = 0; i < employees.size(); i++)
                if (employees.at(i)->getCoord() == clickPoint)
                {

```

## 8. Недопустимо использовать краткие названия

```

int w, h;
w = QDialog::getInt(this, "Ввод параметров", "Введите ширину (минимум 20)", 20, 20);
h = QDialog::getInt(this, "Ввод параметров", "Введите высоту (минимум 20)", 20, 20);

```

## 9. Несоблюдение соглашений о именовании констант

```

#define color_wall QColor(40, 40, 40)
#define color_table QColor(128, 64, 0)
#define color_counter QColor(239, 228, 176)

```

Оценка: 3 из 10