**mainWidget.py**

**from** PyQt5.Qt **import** QWidget, QColor, QPixmap, QIcon, QSize, QCheckBox  
**from** PyQt5.QtWidgets **import** QHBoxLayout, QVBoxLayout, QPushButton, QSplitter, \  
 QComboBox, QLabel, QSpinBox, QFileDialog  
**from** PaintBoard **import** PaintBoard  
  
**class** MainWidget(QWidget):  
 **def** \_\_init\_\_(self, Parent=**None**):  
 super().\_\_init\_\_(Parent)  
 self.\_\_InitData() *# 先初始化数据，再初始化界面* self.\_\_InitView()  
  
 *# 初始化成员变量* **def** \_\_InitData(self):  
 self.\_\_paintBoard = PaintBoard(self)  
  
 *# 初始化界面* **def** \_\_InitView(self):  
 self.setFixedSize(640, 480)  
 self.setWindowTitle(**"生态模拟仿真系统"**)  
  
 *# 新建一个水平布局作为本窗体的主布局* main\_layout = QHBoxLayout(self)  
 *# 设置主布局内边距以及控件间距为10px* main\_layout.setSpacing(10)  
 *# 在主界面左侧放置动画显示界面* main\_layout.addWidget(self.\_\_paintBoard)  
 *# 新建垂直子布局用于放置按键* sub\_layout = QVBoxLayout()  
 *# 设置此子布局和内部控件的间距为10px* sub\_layout.setContentsMargins(10, 10, 10, 10)  
  
 *# 添加文字和按钮  
 # 设置小草数量* self.\_\_label\_Grass = QLabel(self)  
 self.\_\_label\_Grass.setText(**"小草数量"**)  
 self.\_\_label\_Grass.setFixedHeight(20)  
 sub\_layout.addWidget(self.\_\_label\_Grass) *# 文字* self.\_\_spinBox\_Grass = QSpinBox(self)  
 self.\_\_spinBox\_Grass.setMaximum(35) *# 最大值为35* self.\_\_spinBox\_Grass.setMinimum(2) *# 最小值为2* self.\_\_spinBox\_Grass.setValue(10) *# 默认数值为10* self.\_\_spinBox\_Grass.setSingleStep(2) *# 最小变化值为2* self.\_\_spinBox\_Grass.valueChanged.connect(self.on\_GrassNumChange) *# 关联spinBox值变化信号和函数on\_GrassNumChange* sub\_layout.addWidget(self.\_\_spinBox\_Grass) *# 下拉选择框  
  
 # 设置大型植物数量* self.\_\_label\_Macrophyte = QLabel(self)  
 self.\_\_label\_Macrophyte.setText(**"大型植物数量"**)  
 self.\_\_label\_Macrophyte.setFixedHeight(20)  
 sub\_layout.addWidget(self.\_\_label\_Macrophyte)  
  
 self.\_\_spinBox\_Macrophyte = QSpinBox(self)  
 self.\_\_spinBox\_Macrophyte.setMaximum(25)*#最大值为25* self.\_\_spinBox\_Macrophyte.setMinimum(5)*#最小值为5* self.\_\_spinBox\_Macrophyte.setValue(5) *# 默认数值为5* self.\_\_spinBox\_Macrophyte.setSingleStep(2) *# 最小变化值为2* self.\_\_spinBox\_Macrophyte.valueChanged.connect(self.on\_MacrophyteNumChange)  
 sub\_layout.addWidget(self.\_\_spinBox\_Macrophyte)  
  
 *# 设置兔子数量* self.\_\_label\_Rabbit = QLabel(self)  
 self.\_\_label\_Rabbit.setText(**"兔子数量"**)  
 self.\_\_label\_Rabbit.setFixedHeight(20)  
 sub\_layout.addWidget(self.\_\_label\_Rabbit)  
  
 self.\_\_spinBox\_Rabbit = QSpinBox(self)  
 self.\_\_spinBox\_Rabbit.setMaximum(15)*#最大值为15* self.\_\_spinBox\_Rabbit.setMinimum(0)*#最小值为0* self.\_\_spinBox\_Rabbit.setValue(1) *# 默认数值为1* self.\_\_spinBox\_Rabbit.setSingleStep(1) *# 最小变化值为1* self.\_\_spinBox\_Rabbit.valueChanged.connect(self.on\_RabbitNumChange)  
 sub\_layout.addWidget(self.\_\_spinBox\_Rabbit)  
  
 *# 设置蛇数量* self.\_\_label\_Snake = QLabel(self)  
 self.\_\_label\_Snake.setText(**"蛇数量"**)  
 self.\_\_label\_Snake.setFixedHeight(20)  
 sub\_layout.addWidget(self.\_\_label\_Snake)  
  
 self.\_\_spinBox\_Snake = QSpinBox(self)  
 self.\_\_spinBox\_Snake.setMaximum(15)*#最大值15* self.\_\_spinBox\_Snake.setMinimum(0)*#最小值0* self.\_\_spinBox\_Snake.setValue(1) *# 默认数值为1* self.\_\_spinBox\_Snake.setSingleStep(1) *# 最小变化值为1* self.\_\_spinBox\_Snake.valueChanged.connect(self.on\_SnakeNumChange)  
 sub\_layout.addWidget(self.\_\_spinBox\_Snake)  
  
 *# 设置老虎数量* self.\_\_label\_Tiger = QLabel(self)  
 self.\_\_label\_Tiger.setText(**"老虎数量"**)  
 self.\_\_label\_Tiger.setFixedHeight(20)  
 sub\_layout.addWidget(self.\_\_label\_Tiger)  
  
 self.\_\_spinBox\_Tiger = QSpinBox(self)  
 self.\_\_spinBox\_Tiger.setMaximum(10)  
 self.\_\_spinBox\_Tiger.setMinimum(0)  
 self.\_\_spinBox\_Tiger.setValue(1) *# 默认数值为1* self.\_\_spinBox\_Tiger.setSingleStep(1) *# 最小变化值为1* self.\_\_spinBox\_Tiger.valueChanged.connect(self.on\_TigerNumChange)  
 sub\_layout.addWidget(self.\_\_spinBox\_Tiger)  
  
 *# 确定按钮* self.\_\_btn\_Sure = QPushButton(**"确定"**)  
 self.\_\_btn\_Sure.setParent(self) *# 设置父对象为本界面* self.\_\_btn\_Sure.clicked.connect(self.\_\_paintBoard.Print)*# 关联button点击事件和函数Print* sub\_layout.addWidget(self.\_\_btn\_Sure)  
  
 *# 清空按钮* self.\_\_btn\_Clear = QPushButton(**"清空"**)  
 self.\_\_btn\_Clear.setParent(self) *# 设置父对象为本界面* self.\_\_btn\_Clear.clicked.connect(self.\_\_paintBoard.Clear)  
 sub\_layout.addWidget(self.\_\_btn\_Clear)  
  
 *# 退出按钮* self.\_\_btn\_Quit = QPushButton(**"退出"**)  
 self.\_\_btn\_Quit.setParent(self) *# 设置父对象为本界面* self.\_\_btn\_Quit.clicked.connect(self.Quit)  
 sub\_layout.addWidget(self.\_\_btn\_Quit)  
  
 main\_layout.addLayout(sub\_layout) *# 将子布局加入主布局  
  
 # 获取设置后的生物数量* **def** on\_GrassNumChange(self):  
 GrassNum = self.\_\_spinBox\_Grass.value()  
 self.\_\_paintBoard.GrassNumChange(GrassNum)  
  
 **def** on\_MacrophyteNumChange(self):  
 MacrophyteNum = self.\_\_spinBox\_Macrophyte.value()  
 self.\_\_paintBoard.MacrophyteNumChange(MacrophyteNum)  
  
 **def** on\_RabbitNumChange(self):  
 RabbitNum = self.\_\_spinBox\_Rabbit.value()  
 self.\_\_paintBoard.RabbitNumChange(RabbitNum)  
  
 **def** on\_SnakeNumChange(self):  
 SnakeNum = self.\_\_spinBox\_Snake.value()  
 self.\_\_paintBoard.SnakeNumChange(SnakeNum)  
  
 **def** on\_TigerNumChange(self):  
 TigerNum = self.\_\_spinBox\_Tiger.value()  
 self.\_\_paintBoard.TigerNumChange(TigerNum)  
  
 *# 退出* **def** Quit(self):  
 self.close()