

Class 230306

Spring (File Upload) - 5일차

File Upload

기본 설정

▼ pom.xml

```
<!-- commons-fileupload -->
<dependency>
  <groupId>commons-fileupload</groupId>
  <artifactId>commons-fileupload</artifactId>
  <version>1.5</version>
</dependency>
<!-- commons-io -->
<dependency>
  <groupId>commons-io</groupId>
  <artifactId>commons-io</artifactId>
  <version>2.11.0</version>
</dependency>
```

- commons-fileupload 추가
- commons-io 추가

▼ sevlet-context.xml

```
<!-- File Upload -->
<beans:bean id="multipartResolver" class="org.springframework.web.multipart.commons.CommonsMultipartResolver">
  <beans:property name="maxUploadSize" value="10240000"/>
</beans:bean>
```

Car Shop 적용

▼ CarController.java

```
// -----
@Autowired
private ServletContext servletContext;
// -----
@PostMapping("/admin/add")
public String addCar(CardTO car, Model model) {
    MultipartFile carImage = car.getCarImage();
    String upPath = "/resources/upload/";
    String savePath = servletContext.getRealPath(upPath);

    if (carImage != null && !carImage.isEmpty()) {
        String filename = carImage.getOriginalFilename();
        File saveFile = new File(savePath + "/" + filename);
        if(!saveFile.exists()) {
            saveFile.mkdirs();
        }
    }

    try {
```

```

        carImage.transferTo(saveFile);
    } catch (Exception e) {
        throw new RuntimeException("이미지 업로드 실패");
    }
}
carService.setNewCar(car);
return "redirect: /cars/list";
}
// -----

```

▼ cars.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>

<!DOCTYPE html>
<html>
<%@ include file="../include/header.jsp" %>
<body>
<%@ include file="../include/navbar.jsp" %>
    <div class="alert alert-dark">
        <div class="container">
            <h1>차량 보기</h1>
        </div>
    </div>

    <div class="container">
        <div class="row" align="center">

            <c:forEach items="${carList}" var="car">
                <div class = "col-md-4">
                    <h3>${car.cid}</h3>
                    " height="100px">
                    <p>${car.cname}</p>
                    <p>${car.cprice} 만원</p>
                    <p><a href="<c:url value="/cars/car?id=${car.cid}"/>" class="btn btn-secondary btn-sm">상세보기</a></p>
                </div>

            </c:forEach>

        </div>
    </div>
<%@ include file="../include/footer.jsp" %>
</body>
</html>

```

Python Adv - 5일차

GUI Application (Windows)

- Tkinter
 - Python 기본
- wxPython
 - 잘 사용하지 않음
- PyQt5 (권장)
 - 가장 많이 씀

- 기타 등등
 - 잘 안씀 → PyQt5를 쓰자

PyQt5

- riverbankcomputing 사에서 개발
 - C 언어 용으로 개발되었으나 파이썬 용으로 전환
- 현재 최신 버전은 PyQt6
- Doc

Qt Documentation | Home

 <https://doc.qt.io/>

PyQt5 기본 사용

▼ 윈도우 창 띄우기

```
# 윈도우 기본창 띄우기
from PyQt5.QtWidgets import QApplication, QMainWindow # 윈도우 어플리케이션 제작 모듈
import sys

class MyApp(QMainWindow):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.initUI()
    # 창 기본 설정
    def initUI(self):
        self.setWindowTitle('My App') # 타이틀 설정
        self.show() # 창 띄우기

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    sys.exit(app.exec_())
```

▼ 종료 버튼 연결

```
# 종료 버튼
from PyQt5.QtWidgets import QApplication, QMainWindow, QPushButton # 윈도우 어플리케이션 제작 모듈
from PyQt5.QtCore import QCoreApplication
import sys

class MyApp(QMainWindow):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.initUI()
    # 창 기본 설정
    def initUI(self):
        self.setWindowTitle('My App') # 타이틀 설정

        # 종료 버튼
        quitBtn = QPushButton("quit", self)
        quitBtn.clicked.connect(QCoreApplication.instance().quit)

        self.show() # 창 띄우기

if __name__ == "__main__":
```

```
app = QApplication(sys.argv)
ex = MyApp()
sys.exit(app.exec_())
```

▼ 메뉴바 / 종료 메뉴 연결

```
# 메뉴바 설정
from PyQt5.QtWidgets import QApplication, QMainWindow, QPushButton, QApplication, QAction # 윈도우 어플리케이션 제작 모듈
from PyQt5.QtCore import QCoreApplication
from PyQt5.QtGui import QIcon
import sys

class MyApp(QMainWindow):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.initUI()
    # 창 기본 설정
    def initUI(self):

        exitAction = QAction(QIcon('exit.png'), 'Exit', self) # exit메뉴를 설정
        exitAction.setShortcut('ctrl + Q') # 단축키 설정
        exitAction.setStatusTip('Exit Application') # 툴팁 내용 설정
        exitAction.triggered.connect(QApp.quit) # 동작 연결

        self.statusBar() # 상태바 설정 (하단)

        menubar = self.menuBar() # 메뉴바 설정 (상단)
        menubar.setNativeMenuBar(False)
        filemenu = menubar.addMenu('&File') # 메뉴바에 메뉴 추가
        filemenu.addAction(exitAction) # 위에서 만든 exit 메뉴 붙임

        self.setWindowTitle('My App') # 타이틀 설정
        self.setGeometry(300, 300, 300, 300) # 창 크기, 위치 설정
        self.show() # 창 띄우기

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    sys.exit(app.exec_())
```

PyQt5 Designer UI 연동

▼ 윈도우 창 띄우기

```
# Designer의 UI 파일 Import
from PyQt5.QtWidgets import * # 윈도우 어플리케이션 제작 모듈
from PyQt5 import uic
import sys

form_class = uic.loadUiType('mywindow.ui')[0]

class MyApp(QMainWindow, form_class):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.setupUi(self) # ui setup

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    ex.show()
    app.exec_()
```

▼ Push Button 연결

```
# Designer의 UI 파일 Import
from PyQt5.QtWidgets import *      # 윈도우 어플리케이션 제작 모듈
from PyQt5 import uic
import sys

form_class = uic.loadUiType('mywindow.ui')[0]

class MyApp(QMainWindow, form_class):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.setupUi(self)          # ui setup
        self.pushButton.clicked.connect(self.btn_clicked)

    def btn_clicked(self):
        print('클릭 확인')

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    ex.show()
    app.exec_()
```

▼ Push Button 2개 연결

```
# Designer의 UI 파일 Import
from PyQt5.QtWidgets import *      # 윈도우 어플리케이션 제작 모듈
from PyQt5 import uic
import sys

form_class = uic.loadUiType('mywindow2.ui')[0]

class MyApp(QMainWindow, form_class):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.setupUi(self)          # ui setup
        self.pushButton_1.clicked.connect(self.btn_clicked_1)
        self.pushButton_2.clicked.connect(self.btn_clicked_2)

    def btn_clicked_1(self):
        print('1 버튼 클릭 확인')

    def btn_clicked_2(self):
        print('2 버튼 클릭 확인')

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    ex.show()
    app.exec_()
```

▼ Radio Button 연결

```
# Designer의 UI 파일 Import
from PyQt5.QtWidgets import *      # 윈도우 어플리케이션 제작 모듈
from PyQt5 import uic
import sys

form_class = uic.loadUiType('mywindow3.ui')[0]

class MyApp(QMainWindow, form_class):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.setupUi(self)          # ui setup
        self.radioButton_1.clicked.connect(self.radioFunction)
        self.radioButton_2.clicked.connect(self.radioFunction)
        self.radioButton_3.clicked.connect(self.radioFunction)
```

```

def radioFunction(self):
    if self.radioButton_1.isChecked():
        print('라디오 1 체크')
    elif self.radioButton_2.isChecked():
        print('라디오 2 체크')
    elif self.radioButton_3.isChecked():
        print('라디오 3 체크')

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    ex.show()
    app.exec_()

```

▼ Check Box 연결

```

# Designer의 UI 파일 Import
from PyQt5.QtWidgets import *      # 윈도우 어플리케이션 제작 모듈
from PyQt5 import uic
import sys

form_class = uic.loadUiType('mywindow4.ui')[0]

class MyApp(QMainWindow, form_class):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.setupUi(self)          # ui setup

        self.checkBox_1.stateChanged.connect(self.checkFunction)
        self.checkBox_2.stateChanged.connect(self.checkFunction)
        self.checkBox_3.stateChanged.connect(self.checkFunction)

        self.checkBox_4.stateChanged.connect(self.checkFunction2)
        self.checkBox_5.stateChanged.connect(self.checkFunction2)
        self.checkBox_6.stateChanged.connect(self.checkFunction2)

    def checkFunction(self):
        if self.checkBox_1.isChecked():
            print('1 체크')
        if self.checkBox_2.isChecked():
            print('2 체크')
        if self.checkBox_3.isChecked():
            print('3 체크')

    def checkFunction2(self):
        if self.checkBox_4.isChecked():
            print('4 체크')
        if self.checkBox_5.isChecked():
            print('5 체크')
        if self.checkBox_6.isChecked():
            print('6 체크')

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    ex.show()
    app.exec_()

```

▼ Label 연결 및 조작

```

# Designer의 UI 파일 Import
from PyQt5.QtWidgets import *      # 윈도우 어플리케이션 제작 모듈
from PyQt5 import uic
import sys

form_class = uic.loadUiType('mywindow5.ui')[0]

class MyApp(QMainWindow, form_class):
    # 생성자 설정
    def __init__(self):

```

```

        super().__init__()
        self.setupUi(self)          # ui setup

        self.pushButton_1.clicked.connect(self.btnFunction1)
        self.pushButton_2.clicked.connect(self.btnFunction2)

    def btnFunction1(self):
        self.label_1.setText('버튼 1이 클릭되었습니다.')

    def btnFunction2(self):
        self.label_1.setText('버튼 2이 클릭되었습니다.')

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    ex.show()
    app.exec_()

```

- 화면 상에서 수정 불가능한 한 줄의 글자 / 크기를 넘어가면 잘림

▼ TextBrowser 연결

```

# Designer의 UI 파일 Import
from PyQt5.QtWidgets import *      # 윈도우 어플리케이션 제작 모듈
from PyQt5 import uic
import sys

form_class = uic.loadUiType('mywindow6.ui')[0]

class MyApp(QMainWindow, form_class):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.setupUi(self)          # ui setup

        self.pushButton_p.clicked.connect(self.btnFunctionP)
        self.pushButton_c.clicked.connect(self.btnFunctionC)
        self.pushButton_st.clicked.connect(self.btnFunctionST)
        self.pushButton_at.clicked.connect(self.btnFunctionAT)

    def btnFunctionP(self):
        print(self.textBrowser.toPlainText())

    def btnFunctionC(self):
        self.textBrowser.setText('')

    def btnFunctionST(self):
        self.textBrowser.setText('텍스트가 설정되었습니다.\n다음줄\n그 다음줄\n')

    def btnFunctionAT(self):
        self.textBrowser.setText(self.textBrowser.toPlainText() + ' 붙이기')

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    ex.show()
    app.exec_()

```

- 여러 줄의 긴 데이터를 보여줌

▼ Line Edit 연결

```

# Designer의 UI 파일 Import
from PyQt5.QtWidgets import *      # 윈도우 어플리케이션 제작 모듈
from PyQt5 import uic
import sys

form_class = uic.loadUiType('mywindow7.ui')[0]

```

```

class MyApp(QMainWindow, form_class):
    # 생성자 설정
    def __init__(self):
        super().__init__()
        self.setupUi(self)          # ui setup

        self.lineEdit.textChanged.connect(self.leFunction)
        self.pushButton_ct.clicked.connect(self.btnFunctionCT)
        self.pushButton_p.clicked.connect(self.btnFunctionP)

    def leFunction(self):
        print(self.lineEdit.text())

    def btnFunctionP(self):
        self.label.setText(self.lineEdit.text())

    def btnFunctionCT(self):
        self.lineEdit.setText('텍스트 입력')

if __name__ == "__main__":
    app = QApplication(sys.argv)
    ex = MyApp()
    ex.show()
    app.exec_()

```

사용한 위젯 이벤트 정리

- buttons
 - clicked
 - stateChanged
- inputs
 - textChanged