Package main

```
import "GoMS1/server"
```

Overview ▾

This is the server program that serves a course information microservice

Index ▾

Package files

adder.go downloader.go helper.go main.go remover.go returner.go sql.go update.go

Variables

```
var DB *sql.DB
```

func AddNewCourseHandler

```
func AddNewCourseHandler(w http.ResponseWriter, r *http.Request)
```

AddNewCourseHandler add a new course.

func AddNewMaterialHandler

```
func AddNewMaterialHandler(w http.ResponseWriter, r *http.Request)
```

AddNewMaterialHandler adds a new material to the specified course.

func DownloadAllMaterialsHandler

```
func DownloadAllMaterialsHandler(w http.ResponseWriter, r *http.Request)
```

DownloadAllMaterialsHandler downloads files of all materials of the specified course.

func DownloadSpecificMaterialHandler

```
func DownloadSpecificMaterialHandler(w http.ResponseWriter, r *http.Request)
```

DownloadSpecificMaterialHandler downloads the specified material.

func ExecQuery

```
func ExecQuery(w http.ResponseWriter, q string, s ...interface{}) error
```

ExecQuery helps process DB.Exec

func QueryAll

```
func QueryAll(w http.ResponseWriter, i interface{}, q string, s
...interface{}) error
```

QueryAll helps process DB.Query

func RemoveSpecificCourseHandler

```
func RemoveSpecificCourseHandler(w http.ResponseWriter, r *http.Request)
```

RemoveSpecificCourseHandler deletes the specified course.

func **RemoveSpecificMaterialHandler**

```
func RemoveSpecificMaterialHandler(w http.ResponseWriter, r *http.Request)
```

RemoveSpecificMaterialHandler deletes the specified material.

func **ReturnAllCoursesHandler**

```
func ReturnAllCoursesHandler(w http.ResponseWriter, r *http.Request)
```

ReturnAllCoursesHandler returns a list of all courses.

func **ReturnAllMaterialsHandler**

```
func ReturnAllMaterialsHandler(w http.ResponseWriter, r *http.Request)
```

ReturnAllMaterialsHandler returns metadata of all materials of the specified course.

func **ReturnSpecificCourseHandler**

```
func ReturnSpecificCourseHandler(w http.ResponseWriter, r *http.Request)
```

ReturnSpecificCourseHandler returns the specified course.

func **ReturnSpecificMaterialHandler**

```
func ReturnSpecificMaterialHandler(w http.ResponseWriter, r *http.Request)
```

ReturnSpecificMaterialHandler returns the specified material.

func **UpdateSpecificCourseHandler**

```
func UpdateSpecificCourseHandler(w http.ResponseWriter, r *http.Request)
```

UpdateSpecificCourseHandler updates the specified course. You cannot change the COURSE_ID. If you really need to, delete and add new course again.

func **UpdateSpecificMaterialHandler**

```
func UpdateSpecificMaterialHandler(w http.ResponseWriter, r *http.Request)
```

UpdateSpecificMaterialHandler updates the specified material. You cannot change the MATERIAL_ID. If you really need to, delete and add new material again.

func **authenticationMiddleware**

```
func authenticationMiddleware(next http.Handler) http.Handler
```

func **formFileToDisk**

```
func formFileToDisk(w http.ResponseWriter, r *http.Request, v string, m Material) error
```

formFileToDisk saves form file to disk

func **main**

```
func main()
```

main function starts this program

func **receiveJSON**

```
func receiveJSON(r *http.Request) (Course, Material)
```

receiveJSON process receiving incoming json data

func **removeLoc**

```
func removeLoc(loc string) error
```

removeLoc removes a file

func **rootHandler**

```
func rootHandler(w http.ResponseWriter, r *http.Request)
```

rootHandler returns the index page

func **setupDB**

```
func setupDB() func()
```

setupDB detects the environment and connects to the db accordingly

func **setupLog**

```
func setupLog() func()
```

setupLog sets up simple logging. Log file is given a different name everyday.

func **zipUpMaterials**

```
func zipUpMaterials(code string)
```

zipUpMaterials zips up the COURSE_ID folder

type **Course**

```
type Course struct {
    Code        string `json:"code"`
    Title       string `json:"title"`
    Description string `json:"description"`
}
```

type **Material**

```
type Material struct {
    Id       string `json:"id"`
    Sequence int    `json:"sequence"`
    FileName string `json:"filename"`
}
```