

Package api

```
import "GS5Group3/api"
```

[Overview](#)[Index](#)

Overview ▾

Package api is an API server that runs independent of any client.

Index ▾

```
func DeletePlot(db *sql.DB, plotID string)
func DeleteRecord(res http.ResponseWriter, req *http.Request)
func EditPlotAddress(db *sql.DB, plotID string, address string)
func EditPlotVenueName(db *sql.DB, plotID string, venueName string)
func GetAllPlots(w http.ResponseWriter, r *http.Request)
func GetAllUsers(res http.ResponseWriter, req *http.Request)
func InsertPlot(db *sql.DB, p Plot)
func OpenUserDB() *sql.DB
func OpenVenueDB() *sql.DB
func PlotHandler(w http.ResponseWriter, r *http.Request)
func StartServer()
type Plot
type PlotMap
type Users
type VenueMap
```

Package files

api.go bookingHandler.go plotsDB.go plotsVenueHandler.go userDB.go userHandler.go

func DeletePlot

```
func DeletePlot(db *sql.DB, plotID string)
```

func DeleteRecord

```
func DeleteRecord(res http.ResponseWriter, req *http.Request)
```

func EditPlotAddress

```
func EditPlotAddress(db *sql.DB, plotID string, address string)
```

func EditPlotVenueName

```
func EditPlotVenueName(db *sql.DB, plotID string, venueName string)
```

func GetAllPlots

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

/api/v1/plots runs GetAllPlots (GET) Gets all plots and returns it as a json with value {PlotID:{PlotID, VenueName, Address}}

func GetAllUsers

```
func GetAllUsers(res http.ResponseWriter, req *http.Request)
```

func InsertPlot

```
func InsertPlot(db *sql.DB, p Plot)
```

func OpenUserDB

```
func OpenUserDB() *sql.DB
```

func OpenVenueDB

```
func OpenVenueDB() *sql.DB
```

func PlotHandler

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

/api/v1/plots/{plotId} runs PlotHandler (GET, DELETE, POST, PUT) PlotHandler handles GET, DELETE, POST and PUT for individual PlotIDs: GET returns a json {PlotID:{PlotID, VenueName, Address}} based off what PlotID was given. DELETE will delete the respective PlotID given to the handler if it exists. POST will add a new PlotID, VenueName and Address depending on what plotid was given to it. PUT will either add a new PlotID if it does not exist, or alter the VenueName and Address depending on what the administrator fills it with.

func StartServer

```
func StartServer()
```

type Plot

```
type Plot struct {  
    PlotID    string `json:"PlotID"`  
    VenueName string `json:"VenueName"`  
    Address   string `json:"Address"`  
}
```

```
}
```

type PlotMap

```
type PlotMap map[string]Plot
```

type Users

```
type Users struct {  
    Name      string `field:"Name"`  
    Username  string `field:"Username"`  
    Email     string `field:"Email"`  
}
```

type VenueMap

```
type VenueMap map[string]string
```

Build version go1.18.2.

Except as [noted](#), the content of this page is licensed under the Creative Commons Attribution 3.0 License, and code is licensed under a [BSD license](#).

[Terms of Service](#) | [Privacy Policy](#)