GoDoc

Search

Package venue

```
import "goInAction2Assignment/venue"
```

Overview
Index

Overview ▾

Package venue store information of all the function rooms available. It also store the slot availability of a venue in hour. User of the application can view the venue availability using the functions in this package. Any modification of the availability of the slot can be performed using the function in this package as well.

Index ▾

Package files

browse.go search.go update.go

Variables

Rooms available to pick from are stored in slice of struct data type. This slice contains all the details information of the function room such as name, kind, capacity and slot availability.

```
var (
    Room = []roomInfo{
        {Name: "MR01", Kind: "MeetingRoom", Capacity: 10, Slot: TotalAvailableSlot[:279]},
        {Name: "MR02", Kind: "MeetingRoom", Capacity: 10, Slot: TotalAvailableSlot[279:558]},
        {Name: "MR03", Kind: "MeetingRoom", Capacity: 10, Slot: TotalAvailableSlot[558:837]},
        {Name: "MR04", Kind: "MeetingRoom", Capacity: 20, Slot: TotalAvailableSlot[837:1116]},
        {Name: "MR05", Kind: "MeetingRoom", Capacity: 20, Slot: TotalAvailableSlot[1116:1395]},
        {Name: "AR06", Kind: "ActivityRoom", Capacity: 50, Slot: TotalAvailableSlot[1395:1674]},
        {Name: "AR07", Kind: "ActivityRoom", Capacity: 50, Slot: TotalAvailableSlot[1674:1953]},
        {Name: "AR08", Kind: "ActivityRoom", Capacity: 50, Slot: TotalAvailableSlot[1953:2232]},
        {Name: "AD09", Kind: "Auditorium", Capacity: 100, Slot: TotalAvailableSlot[2232:2511]},
        {Name: "AD10", Kind: "Auditorium", Capacity: 100, Slot: TotalAvailableSlot[2511:2790]},
    }
    availableSlot      = make(SlotArr, 0, 279)
    TotalAvailableSlot = make(SlotArr, 0, 2790)
)
```

```
var (
    wg           sync.WaitGroup
    err          error
    mapBookingID = map[string][]int{} //mapBookingID store the results from checkCriteria function.
)
```

AvailableVmap store all the search results for a particular user.

```
var AvailableVmap = map[string][]VenueDetail{}
```

DayOfMonth assign days to different months Not really useful at this stage of project,setup upfront for possible future expansion

```
var DayOfMonth = map[string]int{
    "January": 31, "February": 28, "March": 31, "April": 30, "May": 31, "June": 30,
    "July": 31, "August": 31, "September": 30, "October": 31, "November": 30, "December": 31,
}
```

```
var tpl *template.Template
```

func BrowseVenue

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

BrowseVenue function show table of slot availabiltiy based on user venue selection.

func CheckCriteria

```
func CheckCriteria(date int, time int, duration int, size int, kind string) ([]int, error)
```

CheckCriteria based on criteria provided, search and sort available slot

func GenerateSlots

```
func GenerateSlots()
```

GenerateSlots initiate slice of slots based on date, month and venue (XXXXXX)

func SearchVenue

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

SearchVenue function enable the user to key in all their criteria in the browser and perform a search on available venue based on the inputs.

func indexOfLargest

```
func indexOfLargest(arr []int, n int) int
```

indexOfLargest return the largest number in the slice.

func init

```
func init()
```

func printTableOfSlot

```
func printTableOfSlot(res http.ResponseWriter, req *http.Request, num int)
```

printTableOfSlot print out the availability of all slots for a particular venue.

func screenSlot

```
func screenSlot(find chan int, slotDetail int, duration int) error
```

ScreenSlot was launched (10 goroutines) to perform slot availability binary search

func selectionSort

```
func selectionSort(arr []int, n int)
```

Selection sort was performed based on room number on the available slots array.

func swap

```
func swap(x *int, y *int)
```

swap perform a swap in location between two variable.

type Slot

Each slot is setup with 2 information, slot details and its availability.

```
type Slot struct {
    Info      int
    Available bool
}
```

type SlotArr

All the slots are store in slice of struct data type.

```
type SlotArr []Slot
```

func (SlotArr) RemoveBookedSlot

```
func (s SlotArr) RemoveBookedSlot(bookings []int) error
```

RemovedBookedSlot help to update the slot back to true when the admin remove the slot.

func (SlotArr) UpdateSlotAvailability

```
func (s SlotArr) UpdateSlotAvailability(bookings []int) error
```

UpdateSlotAvailability help to update the slot to false when the venue is booked.

type VenueDetail

VenueDetails is a data type that store details of user search criteria of venue.

```
type VenueDetail struct {
    Day         int
    Month       int
    Time        int
    Room        int
    Duration    int
    Participant int
}
```

type roomInfo

RoomInfo equipped with basic info of room and also an array of its slot availability.

```
type roomInfo struct {
    Name     string
    Kind     string
    Capacity int
    Slot     SlotArr
}
```

Build version go1.16.