Structures

Exercice 1

```
package main

import "fmt"

type item struct {
    id    int
    name  string
    price int
}

type game struct {
    item
    genre string
}

func main() {
    games := []game{
        {
            item:  item{id: 1, name: "god of war", price: 50},
            genre: "action adventure",
        },
        {
            item:  item{id: 2, name: "x-com 2", price: 40},
            genre: "strategy",
        },
        {
            item:  item{id: 3, name: "minecraft", price: 20},
            genre: "sandbox",
        },
    }

    fmt.Printf("Le magasin propose %d jeux.\n\n", len(games))

    for _, g := range games {
        fmt.Printf("#%-4d: %-15q %-20s %d€\n",
            g.id,
            g.name,
            "("+g.genre+")",
            g.price,
        )
    }
}
```

Exercice 2

```
package main

import (
    "bufio"
```

```
    "fmt"
    "os"
)

type item struct {
    id    int
    name  string
    price int
}

type game struct {
    item
    genre string
}

func main() {
    games := []game{
        {
            item:  item{id: 1, name: "god of war", price: 50},
            genre: "action adventure",
        },
        {
            item:  item{id: 2, name: "x-com 2", price: 40},
            genre: "strategy",
        },
        {
            item:  item{id: 3, name: "minecraft", price: 20},
            genre: "sandbox",
        },
    }

    fmt.Printf("Le magasin propose %d jeux.\n", len(games))

    in := bufio.NewScanner(os.Stdin)
    for {
        fmt.Print(`
Commandes :
> list   : liste tous les jeux
> quit   : quitte

`)
        fmt.Print("Votre choix : ")
        in.Scan()
        fmt.Printf("\n")

        switch in.Text() {
        case "quit":
            fmt.Println("Au revoir !")
            return
        case "list":
            for _, g := range games {
                fmt.Printf("#%-4d: %-15q %-20s %d€\n",
                    g.id,
                    g.name,
                    "("+g.genre+")",
                    g.price,
```

```
                )
            }
        default:
            fmt.Println("Commande inconnue.")
        }

    }
}
```

Exercice 3

```
package main

import (
    "bufio"
    "fmt"
    "os"
    "strconv"
    "strings"
)

type item struct {
    id    int
    name  string
    price int
}

type game struct {
    item
    genre string
}

func main() {
    games := []game{
        {
            item:  item{id: 1, name: "god of war", price: 50},
            genre: "action adventure",
        },
        {
            item:  item{id: 2, name: "x-com 2", price: 40},
            genre: "strategy",
        },
        {
            item:  item{id: 3, name: "minecraft", price: 20},
            genre: "sandbox",
        },
    }

    // Index les jeux par id
    gamesByID := make(map[int]game)
    for _, g := range games {
        gamesByID[g.id] = g
    }

    fmt.Printf("Le magasin propose %d jeux.\n", len(games))
```

```
    in := bufio.NewScanner(os.Stdin)
    for {
        fmt.Print(`
Commandes :
> list   : liste tous les jeux
> id N   : affiche le jeu d'identifiant N
> quit   : quitte

`)
        fmt.Print("Votre choix : ")
        in.Scan()
        fmt.Printf("\n")

        cmd := strings.Fields(in.Text())

        if len(cmd) == 0 {
            // Pas de commande, on continue
            continue
        }

        switch cmd[0] {
        case "quit":
            fmt.Println("Au revoir !")
            return
        case "list":
            for _, g := range games {
                fmt.Printf("#%-4d: %-15q %-20s %d€\n",
                    g.id,
                    g.name,
                    "("+g.genre+")",
                    g.price,
                )
            }
        case "id":
            if len(cmd) != 2 {
                fmt.Println("ID invalide.")
                continue
            }
            id, err := strconv.Atoi(cmd[1])
            if err != nil {
                fmt.Println("ID invalide.")
                continue
            }

            g, ok := gamesByID[id]
            if !ok {
                fmt.Println("Jeu introuvable.")
                continue
            }
            fmt.Printf("#%-4d: %-15q %-20s %d€\n",
                g.id,
                g.name,
                "("+g.genre+")",
                g.price,
            )
```

```
        default:
            fmt.Println("Commande inconnue.")
        }

    }
}
```