

Projet réalisé par :

\*\* Labidi Nour el Houda\*\*

\*\* Souissi Oumaima\*\*

\*\* Ben Aoun Siwar\*\*

\*\* 1 ére année , groupe E\*\*

Année universitaire :

\*\*2021-2022\*\*



Description du projet :

\*Ce projet traite tous les chapitres vus le long du semestre.

\*On a opté pour la gestion d’une agence de voyage et plus précisément le département de réservation des hôtels . On a mis l’accent sur 3 axes :

-caractéristiques de l’hôtel

-offres proposées par l’hôtel

-les Extras

\*A partir de ces données le client commande ses réservations basées sur ses préférences .

\* Les clients de l’agence sont stockés dans un tableau, de même pour les hôtels .A partir de ces données , on affiche le meilleur hôtel de l’année et le meilleur client de l’année .

STRUCTURES :

struct date

{

int jour;

int mois;

int annee;

};

typedef struct date DATE;

---------------------------------------------------------------------------------------

struct offre

{

float reduc ;

DATE datei; // offre valable de la datei à la datef

DATE datef;

};

typedef struct offre OFFRE;

---------------------------------------------------------------------------------------

struct extra

{

float pdp ; //prix demi pension

float pcp; ; //prix pension compléte

float pics; ; //prix all inclusive

float psuite ; //prix suite

};

typedef struct extra EXTRA;

---------------------------------------------------------------------------------------

struct caracteristique

{

int nbpisc; // nombres de piscines non couvertes

int nbpiscc; // nombres de piscines non couvertes

char vuesurm; // 'v' si l’hôtel est vue sur mer et 'f' sinon

char suite ; // 'v' si il y a des suites

float prixlpd; // prix du lit et petit dej ( le minimum)

EXTRA ext;

};

typedef struct caracteristique CARACTERISTIQUE;

---------------------------------------------------------------------------------------

struct hotel

{

char nomh[50];

char adresseh[50];

char ad\_emailh[50];

char faciliteh; //‘v’ si l’hôtel autorise le payement par facilité

int nboffreh; //nombre des offres

OFFRE\* offreh; // tableau dynamique des offres existants

CARACTERISTIQUE carac; // structure comportant les caractéristique de l’hôtel

};

typedef struct hotel HOTEL;

---------------------------------------------------------------------------------------

struct reservation

{

char nom\_hotel[50];

DATE date\_in; //date initiale

DATE date\_fin;

int nbjours;

float facture;

char demipen; // 'v' si le client choisit demi pension et ‘f’ sinon

char pencomp; // 'v' si le client choisit pension complète et ‘f’ sinon

char aics; // 'v' si le client choisit all inclusive et ‘f’ sinon

char suite; //'v' si le client choisit une suite et ‘f’ sinon

};

typedef struct reservation RESERVATION;

---------------------------------------------------------------------------------------

struct client

*{*

*int CIN;*

*char nom [50];*

*char prenom[50];*

*int numtell;*

*char ad\_email[50];*

*int* nbreservation***;*** ; // nombre des réservations

RESERVATION reser[200]; // tableau statique comportant l’historique de toutes les réservations du client

};

typedef struct client CLIENT;

---------------------------------------------------------------------------------------

FONCTIONS :

\*\*saisie\_reservation **:**Cette foction va permettre de saisir tous les détailles d’une réservation en appelant la fonction **facturec**.

\*\*remplir\_reservation**:** procédure qui prend en paramètre un tableau statique de réservation du client et sa taille.

Elle permet de remplir un tableau par une réservation saisie par la fonction **saisie\_reservation**.

\*\*ajouter\_client**:** une procédure qui prend en paramètre un tableau dynamique des clients et sa taille.

Elle permet de mettre a jour le tableau de réservation du client si son CIN existe dans la base de donnée sinon elle ajoute le nouveau client dans le fichiers client après avoir saisir ses données.

\*\*afficher\_reservation**:** c’est une procédure qui prend en paramètre une variable de type réservation et qui permet d’afficher toutes les détailles d’une réservation.

\*\*afficher\_tab\_reservation**:** procédure qui prend en paramètre un tableau statique de réservation et sa taille.

Elle permet d’afficher tous les réservations présentes dans le tableau.

\*\*afficher\_client**:** procédure qui prend en paramètre un tableau dynamique du client , sa taille et une variable de type client et permet d’afficher tous les clients du tableau.

\*\* saisie: fonction qui retourne le nombre des hôtel de type entier.

\*\*Ajouter\_hotel :procédure qui prend en paramètre un hôtel (passage par adresse ) et saisie cet hotel à l’aide de la fonction “ saisie\_hotel “.

\*\* Allocation\_offreh :procédure qui prend en paramètre un hôtel et fait l’allocation de son attribut ‘ offreh’ qui est un tableau dynamique des offres de l’hôtel .

\*\*Saisie\_hotel: fonction qui n’a pas de paramètres et qui retourne un hôtel

\*\*Afficher\_hotel: procédure qui permet d’afficher le tableau hôtel , elle prend en paramètre le tableau d’hôtels et sa taille .

\*\*Afficher : procédure qui permet d’afficher un hôtel donné en paramètre .

\*\*Client\_annee :une fonction qui retourne le meilleur client de l’année .

Elle prend en paramètres le tableau des clients et sa taille. Dans cette fonction on compare les factures des clients . Le client ayant la facture la plus élevée est considéré le meilleur .

\*\*Hotel\_annee : procédure qui permet d’afficher le nom , l'adresse, l’adresse e-mail et le nombre des offres du meilleur hôtel de l’année. Elle prend en paramètres le tableau d’hôtels et sa taille .

Dans cette fonction on compare les nombres des offres des d’Hotels . l’hotel ayant la facture le nombre des offres le plus élevée est considéré le meilleur .