

في مطعم "الأخوة" يمكن حجز مكان لزبون واحد أو أكثر. بعد انتهاء الوجبة، يدفع كل واحد من الزبائن بمفرده مقابل الطعام الذي أكله. يستطيع كل زبون أن يختار الدفع نقداً أو بالتطبيق أو ببطاقة اعتماد. لمن يدفع ببطاقة اعتماد هناك إمكانية لدفع الحساب بعدة أقساط متساوية.

بهدف إدارة منظومة الدفعات، يطوِّرون للمطعم مشروعاً فيه واجهة التطبيق (**IPayment Interface**)، والفئات **Restaurant, Reservation, Credit, App, Cash**، كما هو مفصّل فيما يلي:

❖ **IPayment** – الدفع

توجد في واجهة التطبيق العملية `double getPrice()`

❖ **Cash** – الدفع نقداً

صفئا الفئة:

- `sumCash` – المبلغ للدفع نقداً، من نمط حقيقيّ.
- `name` – اسم الزبون، من نمط نصّ.

❖ **App** – الدفع بالتطبيق

صفئا الفئة:

- `sumApp` – المبلغ للدفع بالتطبيق، من نمط حقيقيّ.
- `phoneNumber` – رقم هاتف، من نمط نصّ.

❖ **Credit** – الدفع ببطاقة اعتماد

صفات الفئة:

- `num` – عدد الأقساط، من نمط صحيح.
- `part` – المبلغ في كل قسط، من نمط حقيقيّ.
- `creditNumber` – رقم بطاقة الاعتماد، من نمط نصّ.

❖ **Reservation** – حجز

صفات الفئة:

- `date` – تاريخ الحجز، من نمط نصّ.
- `total` – الحساب الكليّ لجميع الزبائن معاً في نفس الحجز، من نمط حقيقيّ.
- `payments` – مصفوفة تحفظ المبلغ الذي دفعه كل واحد من الزبائن في نفس الحجز، من نمط **IPayment** (المصفوفة بدون قيم `null`).

❖ **Restaurant** – المطعم

صفئا الفئة:

- `array` – مصفوفة بكبر 10,000، من نمط **Reservation**.
- `current` – كمّية الحجوزات المحفوظة، من نمط صحيح.

افتراضوا أنّ الحجوزات تُحفظ بتسلسل في مصفوفة وليس بينها قيم `null`.

ملاحظة: افتراضوا أنّه في كل فئة تمّ تعريف عمليّتي `get` و `set` وعمليات بنائية.

أ. (1) ارسموا مخططاً هرمياً يصف العلاقات بين فئات وواجهة تطبيق المشروع.

يجب الإشارة إلى تطبيق واجهة التطبيق بواسطة السهم <---- و إلى احتواء بواسطة الإشارة  .

(2) يقدم المطعم تخفيضات للزبائن حسب طريقة الدفع: الزبون الذي يدفع نقداً ومبلغ الدفع هو أكثر من 200 شيكل يستحق تخفيضاً بقيمة 10% ، والزبون الذي يدفع بالتطبيق يستحق دائماً تخفيضاً بقيمة 5% ، والزبون الذي يدفع ببطاقة اعتماد يدفع سعراً كاملاً.

عملية واجهة التطبيق getPrice تُعيد مبلغ الدفع الذي يدفعه كل زبون بعد حساب التخفيض (عندما لا يوجد تخفيض، تُعيد العملية سعراً كاملاً). طبقوا عملية واجهة التطبيق getPrice في كل واحدة من الفئات اللازمة.

ب. اكتبوا في الفئة **Reservation** عملية باسم cashTotal . تُعيد العملية مبلغ المال النقدي الذي تم تلقيه في الحجز (بعد حساب التخفيض).

ج. أمامكم عملية في الفئة **Reservation** :

```
public void printDetails () {  
    for (int i = 0; i < payments.length; i++) {  
        System.out.println (payments[i].getDetails());  
    }  
}
```

تطبع العملية بالنسبة لكل زبون تفاصيل معينة حسب المعايير التالية:

– إذا دفع الزبون نقداً، تطبع العملية اسمه .

– إذا دفع الزبون بالتطبيق، تطبع العملية رقم هاتفه .

– إذا دفع الزبون ببطاقة اعتماد، تطبع العملية رقم بطاقة اعتماده .

أضيفوا العمليات الضرورية إلى المشروع كي تنفذ العملية المطلوب . اذكروا بالنسبة لكل عملية أضفتموها، إلى أية فئة أو إلى أية واجهة تطبيق تتبع .

ملاحظة: لا تغيروا العملية printDetails .