

Soru 1: (20 Puan)

Birinci dosyada bir lokantaya girmiş kişilerin listesi mevcuttur (visitor.txt). İkinci dosyada ise yemek almış kişilerin listesi mevcuttur (customer.txt). Bu listeleri kıyaslanarak (linq kullanılabilir) gün bazında olacak şekilde

- kaç kişinin binaya geldiği
- kaç kişinin yemek yediği ve oranını [(yemek yiyen kişi sayısı *100) / toplam kişi sayısı]
- kaç kişinin yemek yemeden geri döndüğünü ve oranını [(yemek yemeden dönen kişi sayısı * 100) / toplam kişi sayısı]

ekrana yazınız.

(Bu soru daha çok dosya işlemleri üzerine odaklanmıştır. Sınıf kullanma zorunluluğu yoktur)

Soru 2: (30 Puan)

Bu soruda telefon rehberi uygulaması hazırladığınızı düşünebilirsiniz.

- Kişiler bir dosyaya kaydedilerek saklanacaktır.
- Kişiler için: isim, soy isim, ev adresi, notlar ve telefon numarası listesi şeklinde bilgileri tutulacaktır.
- Bu bilgiler isteğe bağlı olarak girilecek olup en az bir alanın dolu olması gerekmektedir.
- Uygulamada yeni kişi ekleme, kişi silme, güncelleme gibi seçenekler olmalıdır.
- Uygulamada kişi arama seçeneği olmalı ve girilen text kullanıcının isminde, soy isminde veya telefon numarası alanlarında aranmalıdır.
 - (Not: %100 eşleşmeye gerek olmamalıdır.
 - Örnek: "ad" yazılması halinde ismi içerisinde "...ad..." olanlar veya soy ismi içerisinde "...ad..." olanlar listelenmelidir.
 - Örnek: "212 555" şeklinde bir aramanın sonucu "2125551122", "0 212 55 51 133" gibi telefon numaralarına sahip olan kişiler olacaktır.
 - Arama sonuçlarında her zaman kullanıcı isimleri (isim soyisim) şeklinde olmalıdır.
 - Bu kişiler listelendikten sonra detaylı olarak incelenmek istenen kişi, kullanıcıya sorularak tespit edilecek ve o kişi hakkındaki bilgiler ekrana basılacaktır.

Soru 3: (60 Puan)

```
// dönüş değeri metre cinsinden

private static
{
    double R = 6371000;

    double dlon = (point1.Longitude - point2.Longitude) * Math.PI / 180;
    double dlat = (point1.Latitude - point2.Latitude) * Math.PI / 180;

    double a = Math.Pow(Math.Sin((dlat / 2)), 2) +
                Math.Cos(point1.Latitude * Math.PI / 180) *
                Math.Cos(point2.Latitude * Math.PI / 180) *
                Math.Pow(Math.Sin((dlon / 2)), 2);

    double c = 2 * Math.Atan2(Math.Sqrt(a), Math.Sqrt(1 - a));
    double distance = R * c;
    return distance;
}
```

Yukarıda iki nokta arasında uzaklığı bulan fonksiyon verilmiştir.

Ekte birkaç nokta ve bu noktaların isimleri yazılı olan bir dosya mevcuttur. Ayrıca yine ekte kişilerin id ve daha önce bulundukları noktaları gösteren bir dosya mevcuttur. Sizden istenen ise şudur:

- **Program çalışmaya başlayınca yapılacaklar:**

1. Nokta'yı temsil edecek bir sınıf tasarlayınız.
2. Bu nokta sınıfına ait nesneleri tutacak bir liste hazırlayınız (places.txt)
3. Kişi'yi temsil edecek bir sınıf hazırlayınız. (id değeri ve geçtiği noktaları tutacak bir liste şeklinde olabilir)
4. Dosya vasıtası ile kişi nesneleri hazırlayınız. (personList.txt)

- **Operasyonlar**

1. Yeni bir kişi oluşturulabilir olmalıdır (yeni bir id değeri otomatik verilecek ve bulunduğu bir konum istenecek).
2. Bir kişi için bulunduğu bir koordinat eklenebilir (id değeri ile kişi tespit edilecek ve bulunduğu konumlar listesine bir konum eklenecek).
3. Konumlar dosyasına yeni bir yer eklenebilir (ismi ve enlem boylamı ile listeye yeni bir yer eklenebilir) veya silinebilir.
4. Kişilerin bulundukları konumları çözümle işlemi olmalıdır. (kişilerin konumları ile konumlar listesi kıyaslanacaktır. Kişi konumları ile konumlar listesi arasındaki noktalardan 100 metreden yakın olan yerler olması halinde kişinin o adreste bulunduğu düşünülecektir ve bu adres ekrana yazılacaktır.

- **Program kapanmadan önce emin olun**

1. Yeni eklenen konumlar places.txt dosyasına eklenmelidir.
2. Kişiler için yeni yerlere gitme durumları eklenince ve gittikleri yerlere ait bilgiler listeden çıkarıldığı zaman personList.txt dosyası güncellenmelidir.