
HMIN210 - Architecture distribuée : TP Soap

Sujet : Création et utilisation de services web.

Date : 12 avril 2021.

Tâches demandées pour ce rendu :

- Réaliser la question 1 de l'exercice 4 du TP.
 - Produire un diagramme UML
- Réaliser la question 2 de l'exercice 4 du TP.
 - Produire un diagramme UML
- Réaliser la question 3 de l'exercice 4 du TP.

Descriptif pour ce rendu :

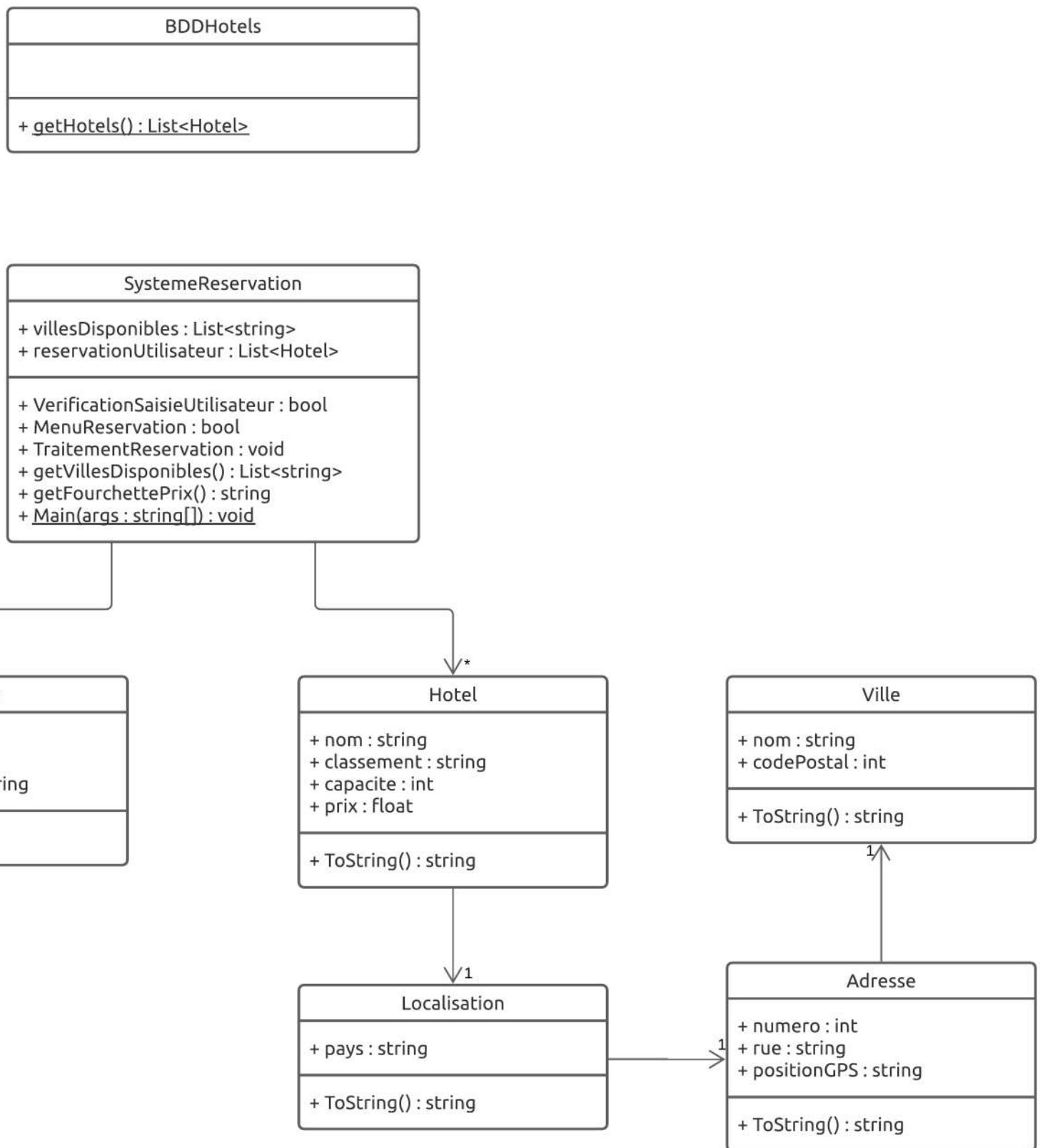
Pour ce rendu nous avons créé deux applications de réservation d'hôtels. La première application n'utilise pas de web service, elle englobe toutes les différentes fonctionnalités au sein de son architecture. La deuxième application utilise deux services web : Consultation et Réservation. Pour chacune des différentes composantes de nos deux applications, des classes ont été créées (hôtels, agences...). Nous avons notamment créé 10 hôtels, tous 5 étoiles afin d'éviter que les critères de recherche de l'utilisateur ne filtre trop rapidement les différents hôtels néanmoins il est aisément possible d'ajouter de nouveaux hôtels au sein la méthode statique *GetHotels()* de la classe *BDDHotels* (voir UML). De la même manière, nous avons créé une classe *BDDAgences* qui contient les agences autorisées à utiliser nos 2 services web.

Nous avons réalisé 2 diagrammes UML, un pour la version non distribuée et l'autre pour la version distribuée. Nous avons également réalisé un diagramme de séquence pour la version distribuée afin de synthétiser les différents échanges qu'il y a entre le système de réservation et les deux services web.

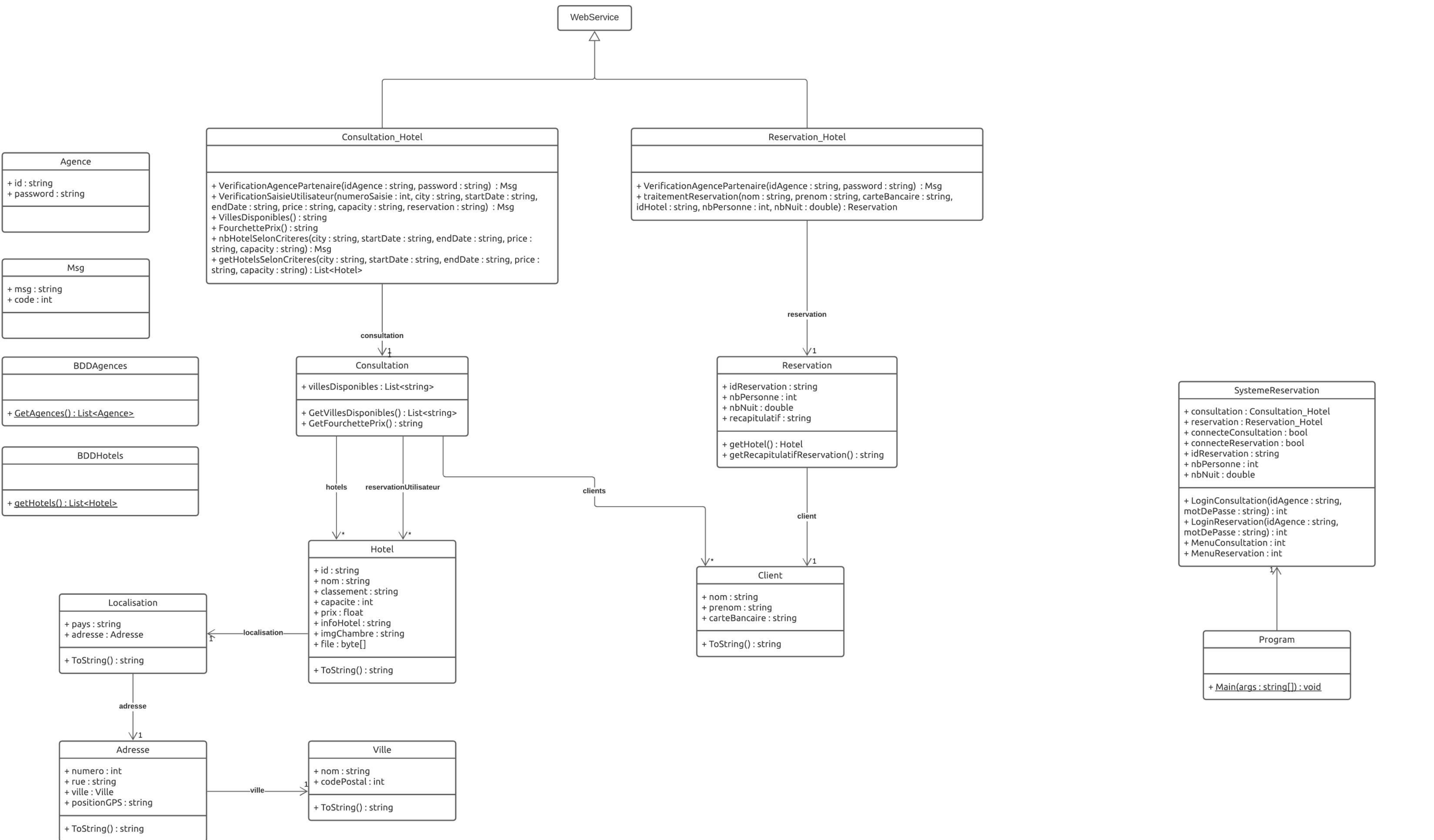
Tous ces schémas sont présents dans les pages suivantes de ce rendu. Est également présent dans les pages suivantes, un cas d'utilisation type de la version distribuée avec des commentaires afin d'illustrer le fonctionnement.

Lien du dépôt : <https://drive.google.com/drive/folders/1mQwoKQHTbkYdondj210dyMfcSw7EzEgn?usp=sharing>

Tous les screens sont présents sur les pages suivantes.



Modèle UML de la version sans distribution



Architecture des services web

Architecture de l'application cliente

Modèle UML de la
version avec distribution

n°Msg	Contenu
1	Listes des villes disponibles ; Dans quelle ville souhaitez-vous réserver ?
2	Quelle est la date d'arrivée ?
3	Quelle est la date de départ ?
4	Les prix appartiennent à cet intervalle ; Quel est votre budget maximum ?
5	Combien de personne faut-il héberger ?
6	Affichage de la liste des Hôtels correspondants à la recherche
7	Renvoi l'image de la réservation sous la forme d'une string base64
8	Demande d'infos sur l'utilisateur (nom, prénom, n° carte bleu...)
9	Obtention d'une référence de la réservation contenant le récapitulatif

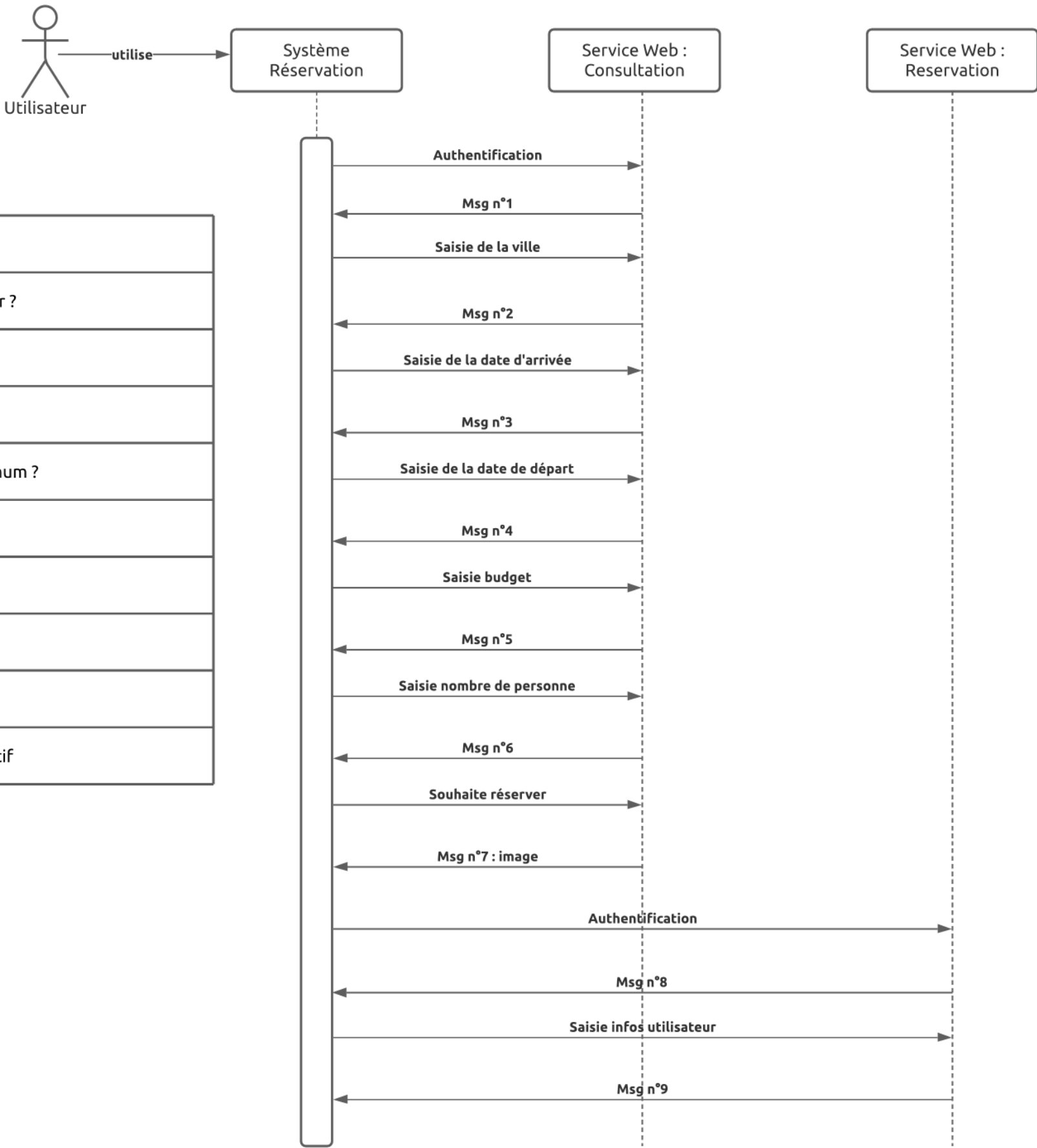
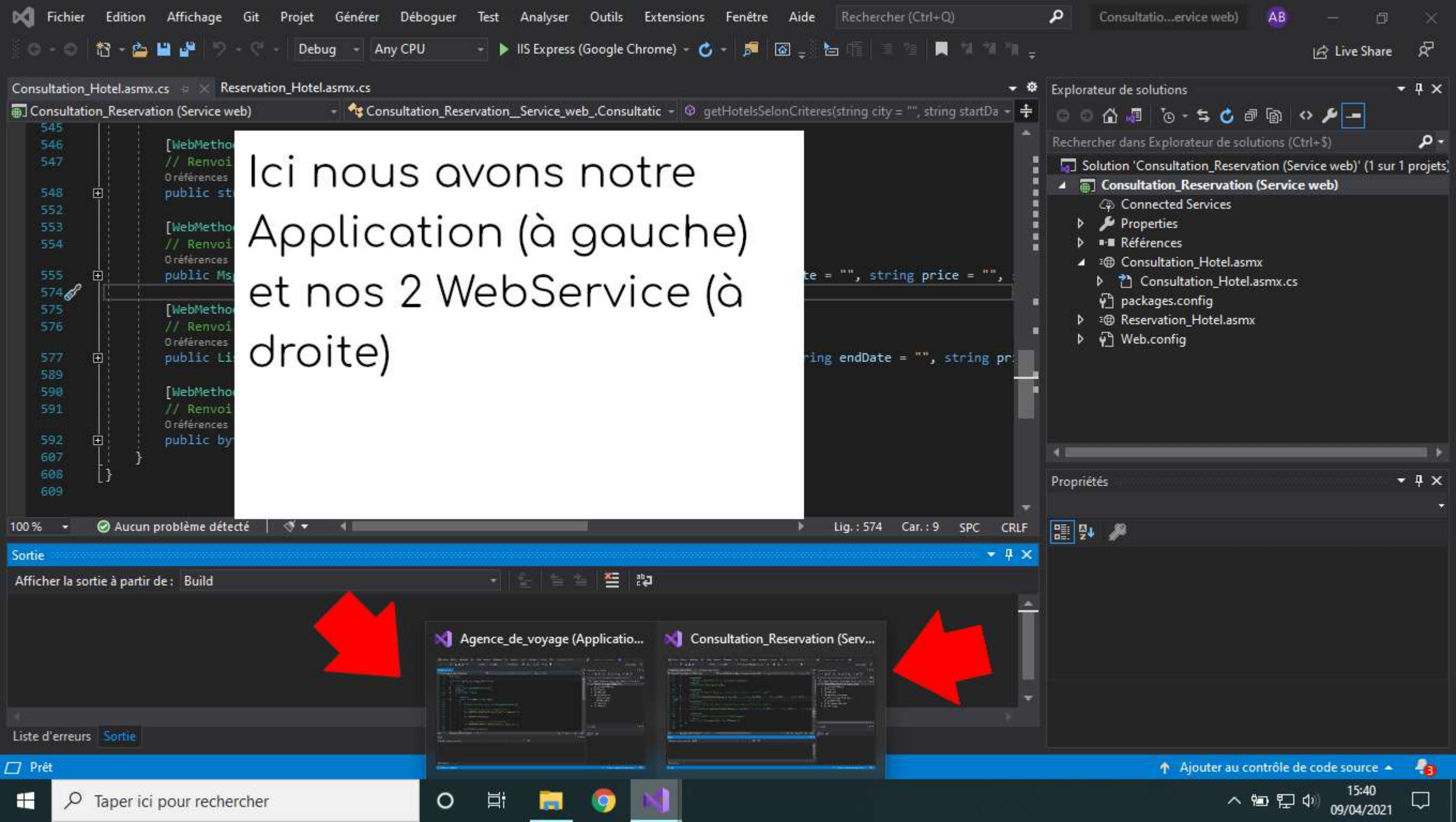
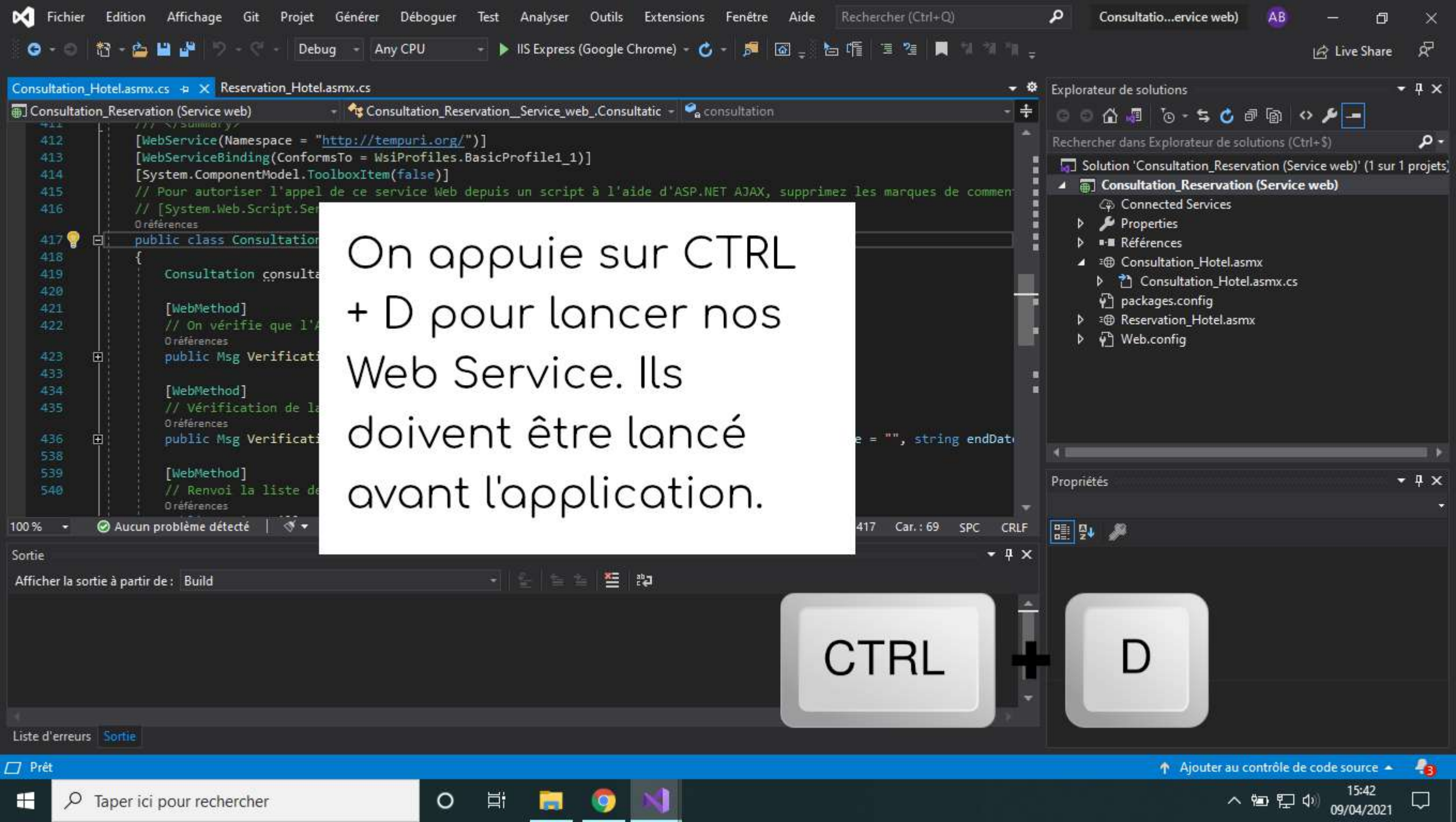


Diagramme de séquences de la version distribuée

Exemple de cas d'utilisation de la version
distribuée



Ici nous avons notre
Application (à gauche)
et nos 2 WebService (à
droite)



On appuie sur CTRL
+ D pour lancer nos
Web Service. Ils
doivent être lancé
avant l'application.

CTRL

D

Consultation_Hotel

Les opérations suivantes sont prises en charge. Pour une définition formelle, prenez connaissance de la [Description du service](#).

- [FourchettePrix](#)
- [GetImageFile](#)
- [VerificationAgencePartenaire](#)
- [VerificationSaisieUtilisateur](#)
- [VillesDisponibles](#)
- [getHotelsSelonCriteres](#)
- [nbHotelSelonCriteres](#)

Ce service web utilise <http://tempuri.org/> comme espace de noms par défaut. **Recommandation : modifiez l'espace de noms par défaut avant que le service**

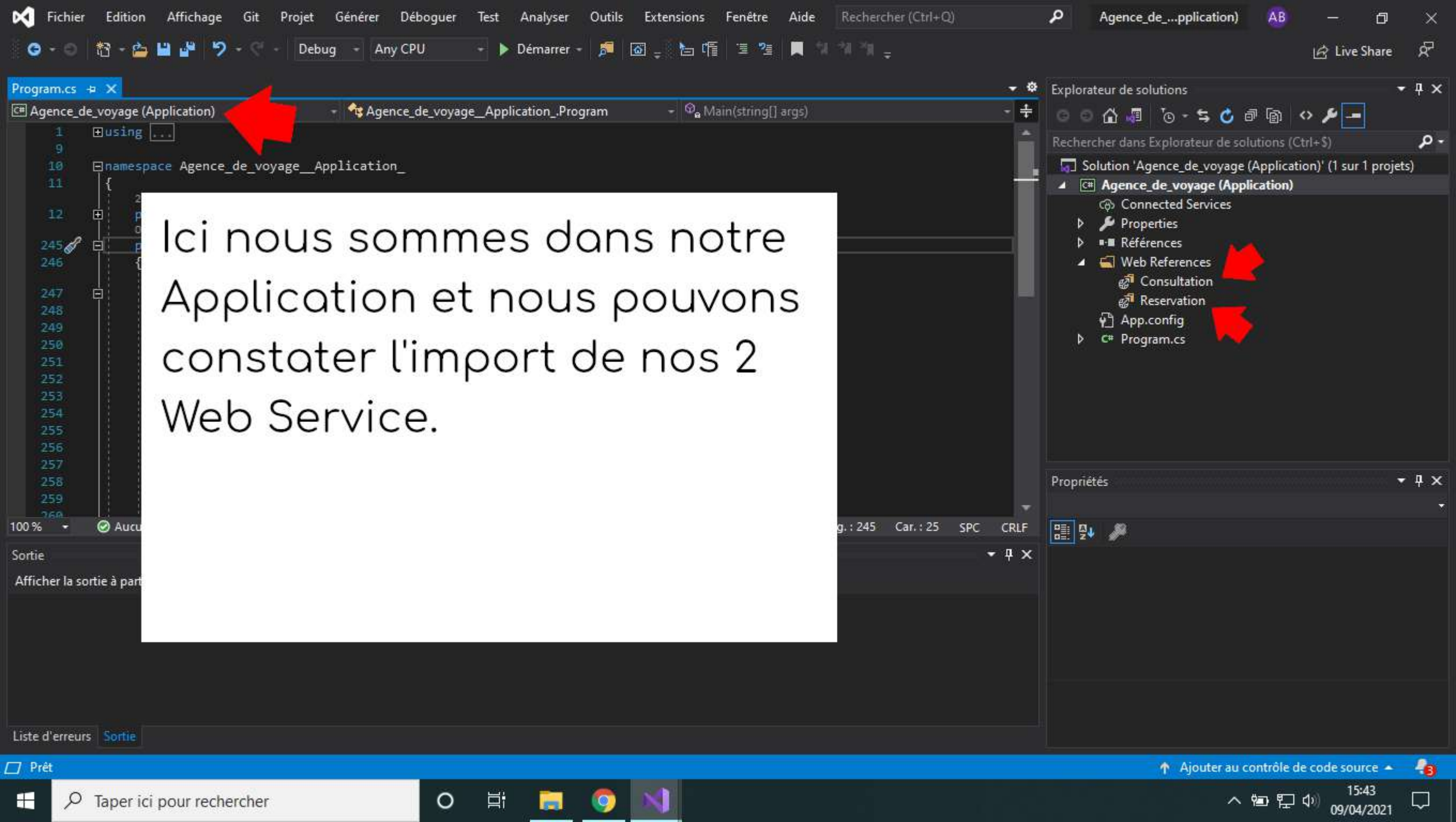
Chaque service web XML a besoin d'être identifié par un espace de noms unique ; ainsi, les cours de développement, cependant, les services web XML publiés doivent utiliser un espace de noms unique. Votre service web XML doit être identifié par un espace de noms que vous contrôlez. Par exemple, les noms de services web XML ressemblent à des URL, ils n'ont pas besoin de pointer vers un service web réel. Pour la création de services web XML à l'aide de ASP.NET, l'espace de noms par défaut peut contenir les méthodes du service Web XML. Vous trouverez ci-dessous un exemple de code C#

```
C#  
[WebService(Namespace="http://microsoft.com/webservices/")]  
public class MyWebService {  
    // implémentation  
}
```

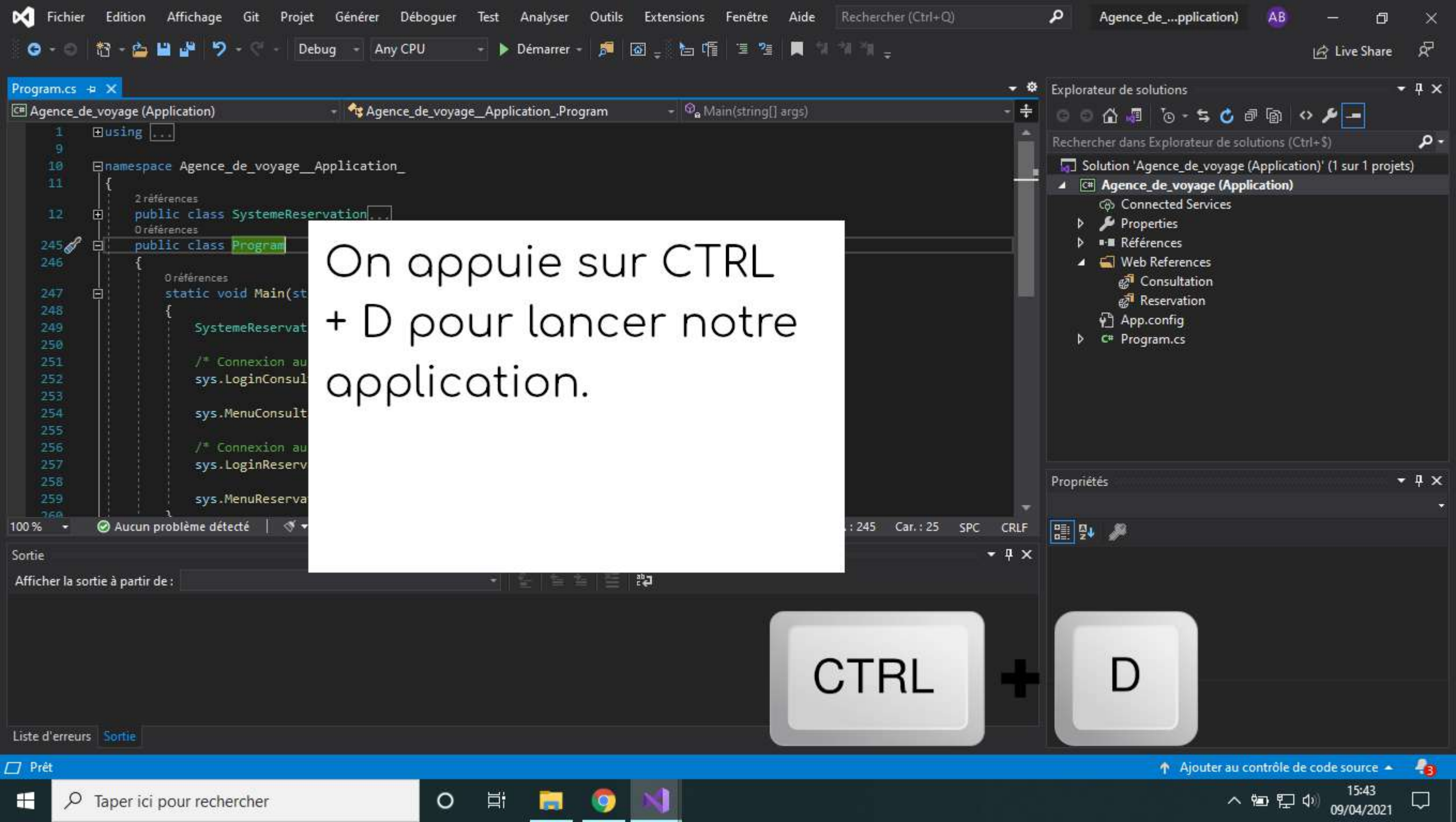
```
Visual Basic  
<WebService(Namespace="http://microsoft.com/webservices/")> Public Class MyWebServ  
    ' implémentation  
End Class
```

```
C++
```

Une fois qu'on a lancé notre WebService, le navigateur se lance sur l'url visible à l'écran. Par ailleurs on peut constater toutes les opérations fournies par notre WebService.

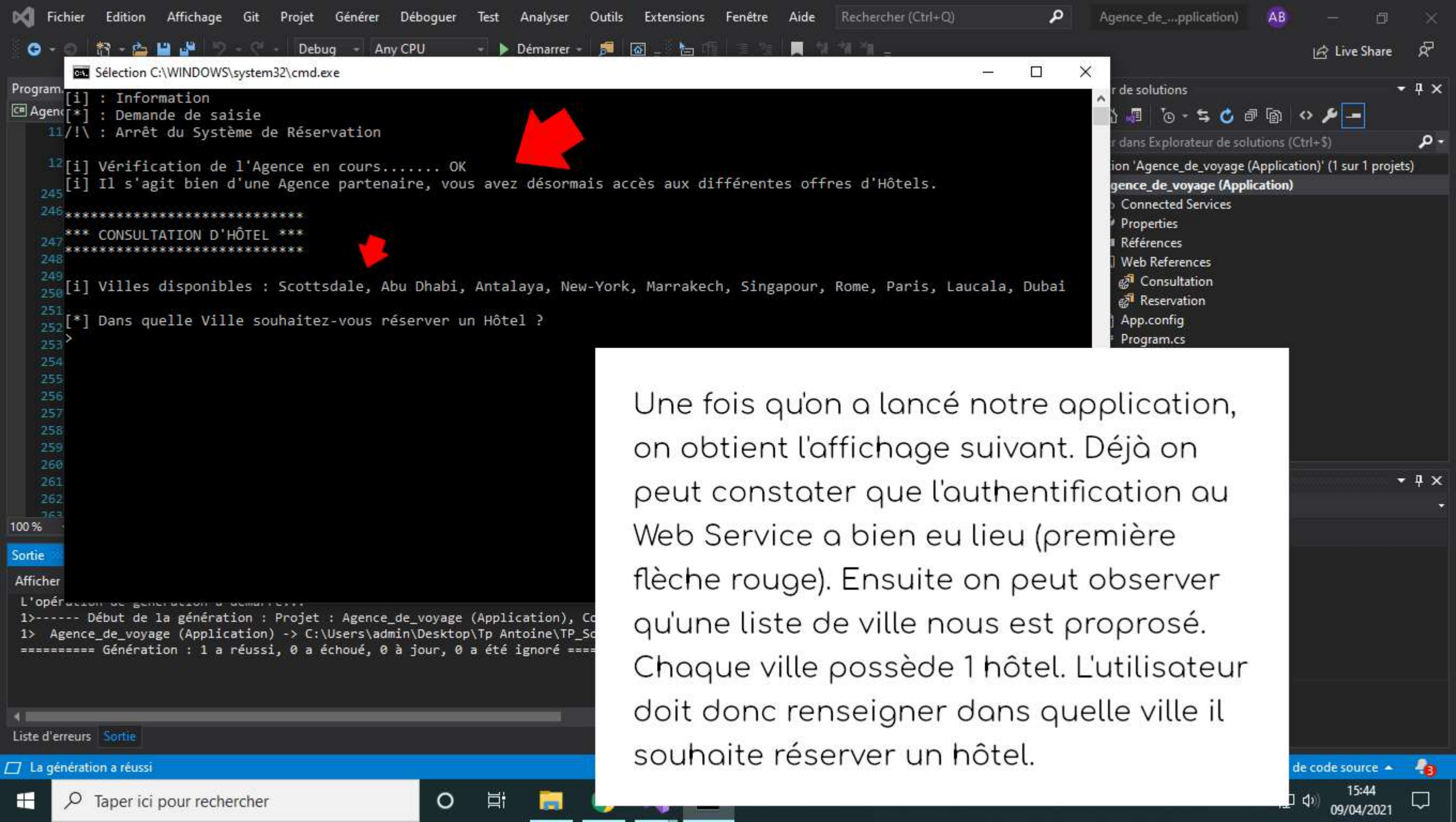


Ici nous sommes dans notre Application et nous pouvons constater l'import de nos 2 Web Service.

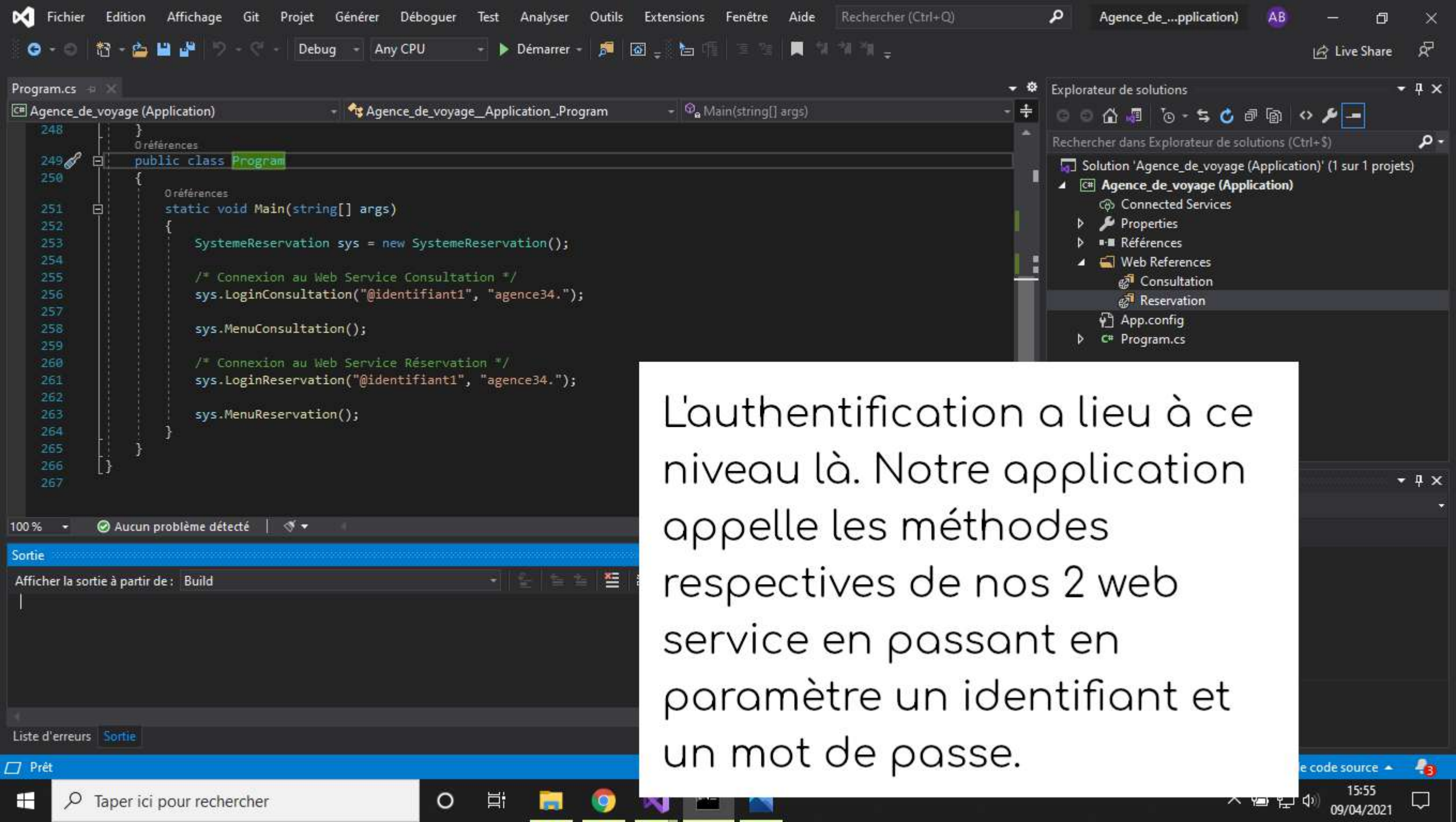


On appuie sur CTRL + D pour lancer notre application.





Une fois qu'on a lancé notre application, on obtient l'affichage suivant. Déjà on peut constater que l'authentification au Web Service a bien eu lieu (première flèche rouge). Ensuite on peut observer qu'une liste de ville nous est proposée. Chaque ville possède 1 hôtel. L'utilisateur doit donc renseigner dans quelle ville il souhaite réserver un hôtel.



L'authentification a lieu à ce niveau là. Notre application appelle les méthodes respectives de nos 2 web service en passant en paramètre un identifiant et un mot de passe.

FichierEditionAffichageGitProjetGénérerDéboguerTestAnalyserOutilsExtensionsFenêtreAideRechercher (Ctrl+Q)Consultatio...ervice web)AB—

DebugAny CPUIIS Express (Google Chrome)

Live Share

Consultation_Hotel.asmx.csReservation_Hotel.asmx.cs

Consultation_Reservation (Service web)Consultation_Reservation_Service_web_Consultatioconsultation

```
1  using ...
11
12  namespace Consultation_Reservation_Service_web_
13  {
14      // Base de donnée contenant les Agences partenaires
15      2 références
16      public class BDDAgences
17      {
18          2 références
19          public static List<Agence> GetAgences()
20          {
21              List<Agence> agences = new List<Agence>();
22
23              agences.Add(new Agence("@identifiant1", "agence34."));
24              agences.Add(new Agence("@identifiant2", "agence75."));
25              agences.Add(new Agence("@identifiant3", "agence69."));
26
27              return agences;
28          }
29
30      // Descriptif Agence
31      7 références
32      public class Agence
```

Explorateur de solutions

Rechercher dans Explorateur de solutions (Ctrl+S)

Solution 'Consultation_Reservation (Service web)' (1 sur 1 projets)

Consultation_Reservation (Service web)

Connected Services

Properties

Références

Consultation_Hotel.asmx

Consultation_Hotel.asmx.cs

packages.config

Reservation_Hotel.asmx

Web.config

100 %

Aucun problème détecté

Sortie

Afficher la sortie à partir de: Build

L'opération de génération a démarré...

1>----- Début de la génération : Projet : Consultation_Reservation (Service web), Conf

1> Consultation_Reservation (Service web) -> C:\Users\admin\Desktop\Tp Antoine\TP_Soap

===== Génération : 1 a réussi, 0 a échoué, 0 à jour, 0 a été ignoré =====

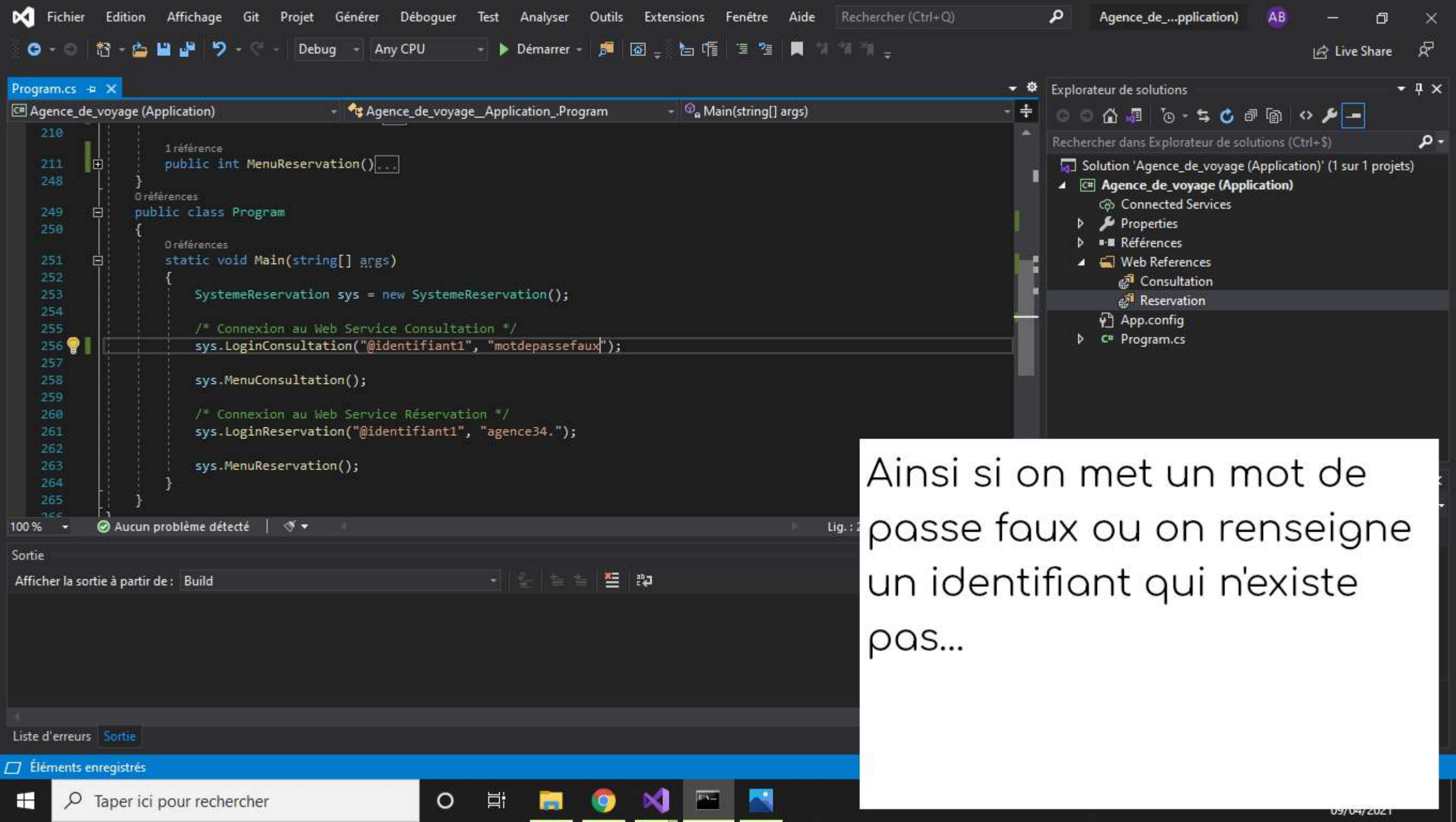
Liste d'erreursSortie

La génération a réussi

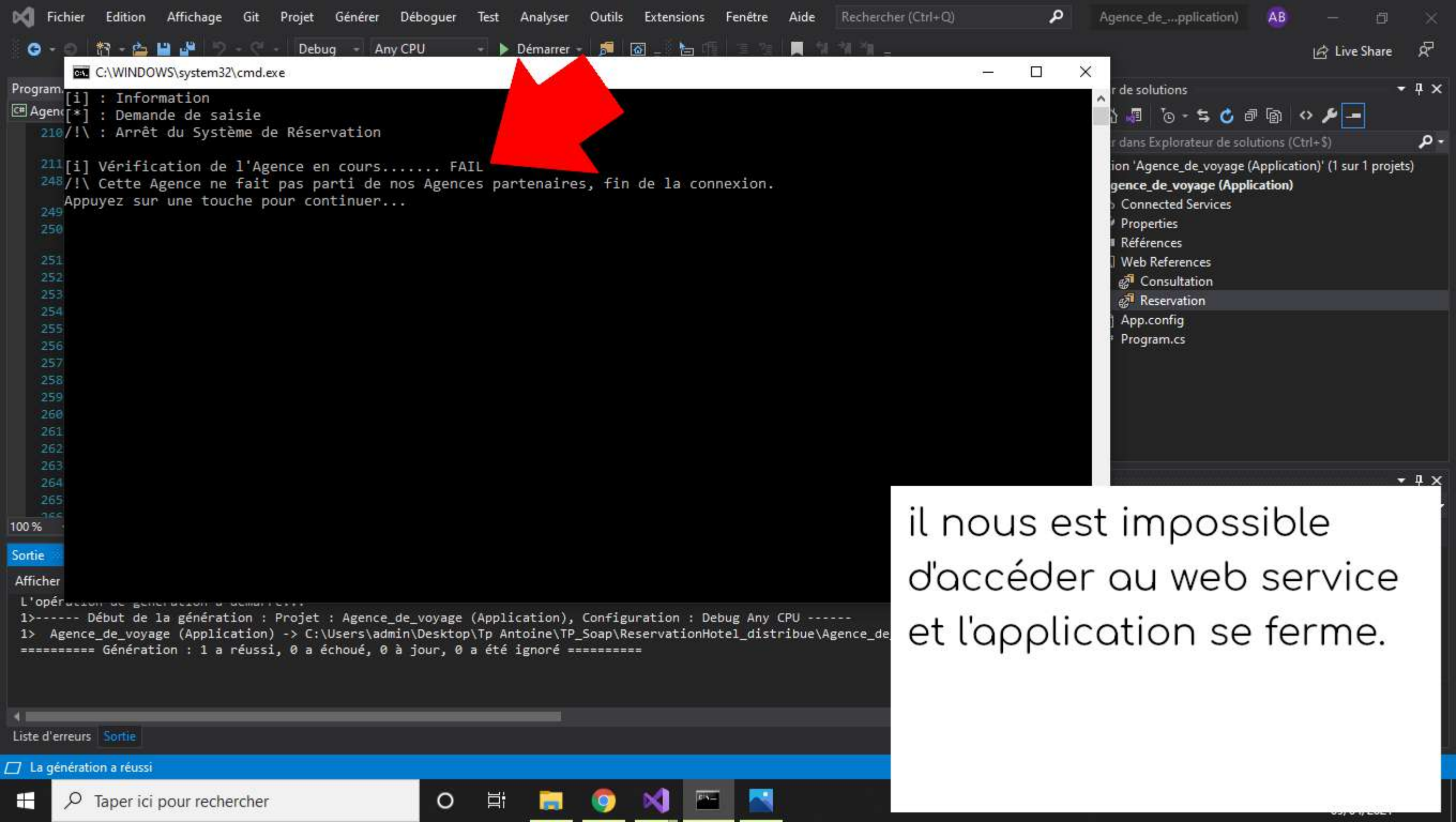
Nos 2 web service partagent le même namespace ou est présent une classe BDDAgences possédant une méthode statique renvoyant la liste des agences autorisées à se connecter.

Taper ici pour rechercher

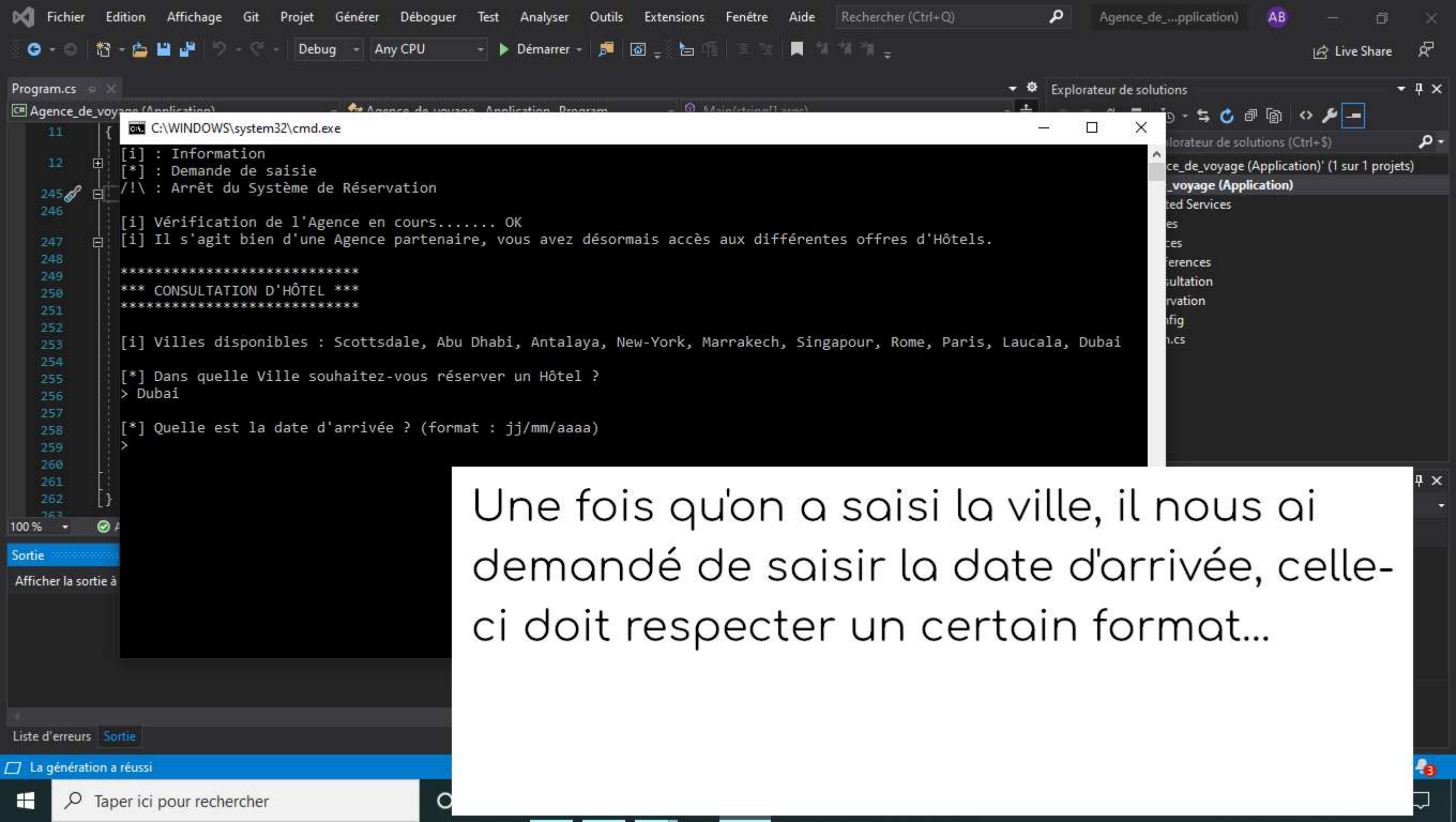
09/04/2021

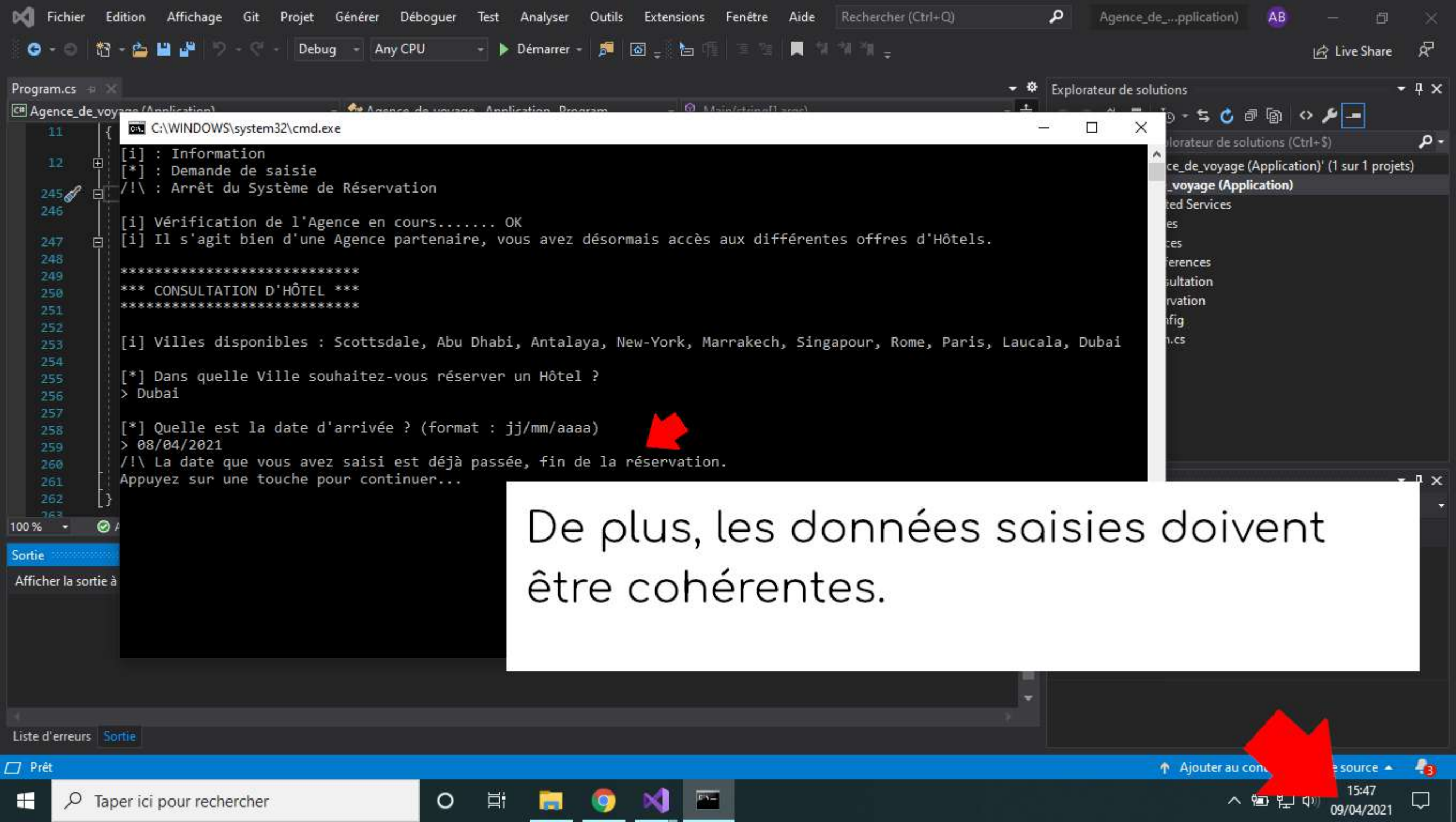


Ainsi si on met un mot de passe faux ou on renseigne un identifiant qui n'existe pas...

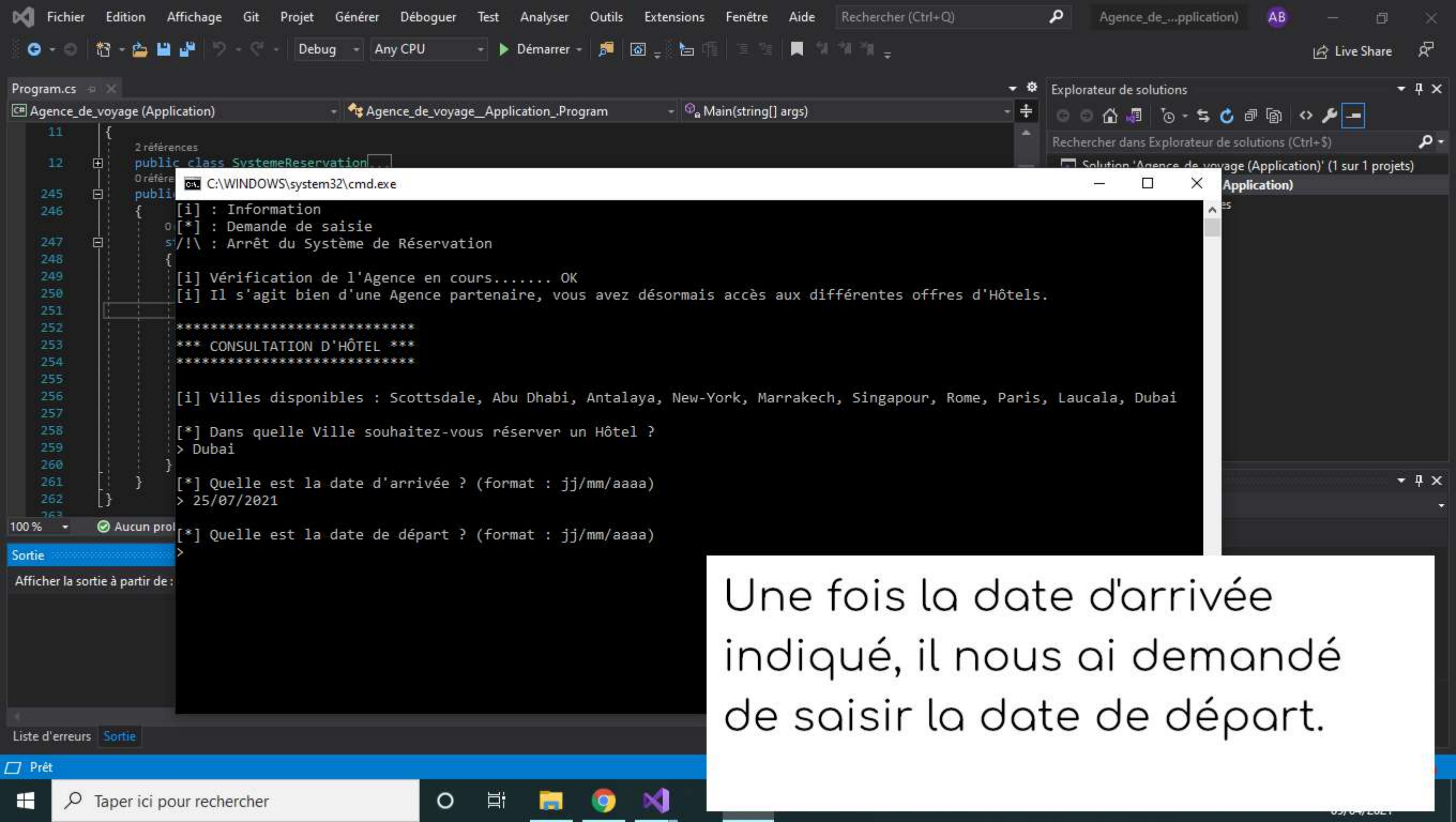


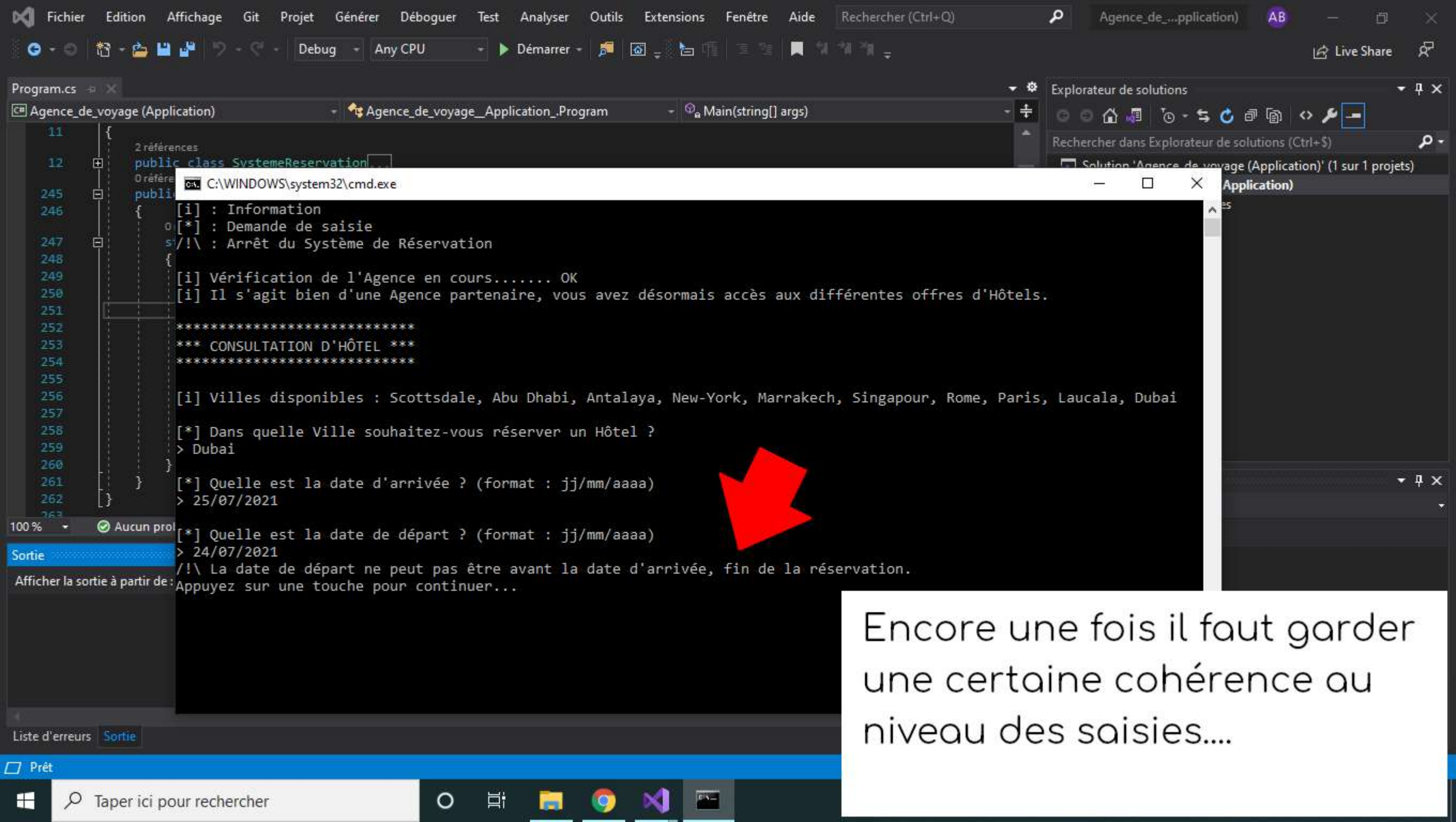
il nous est impossible
d'accéder au web service
et l'application se ferme.

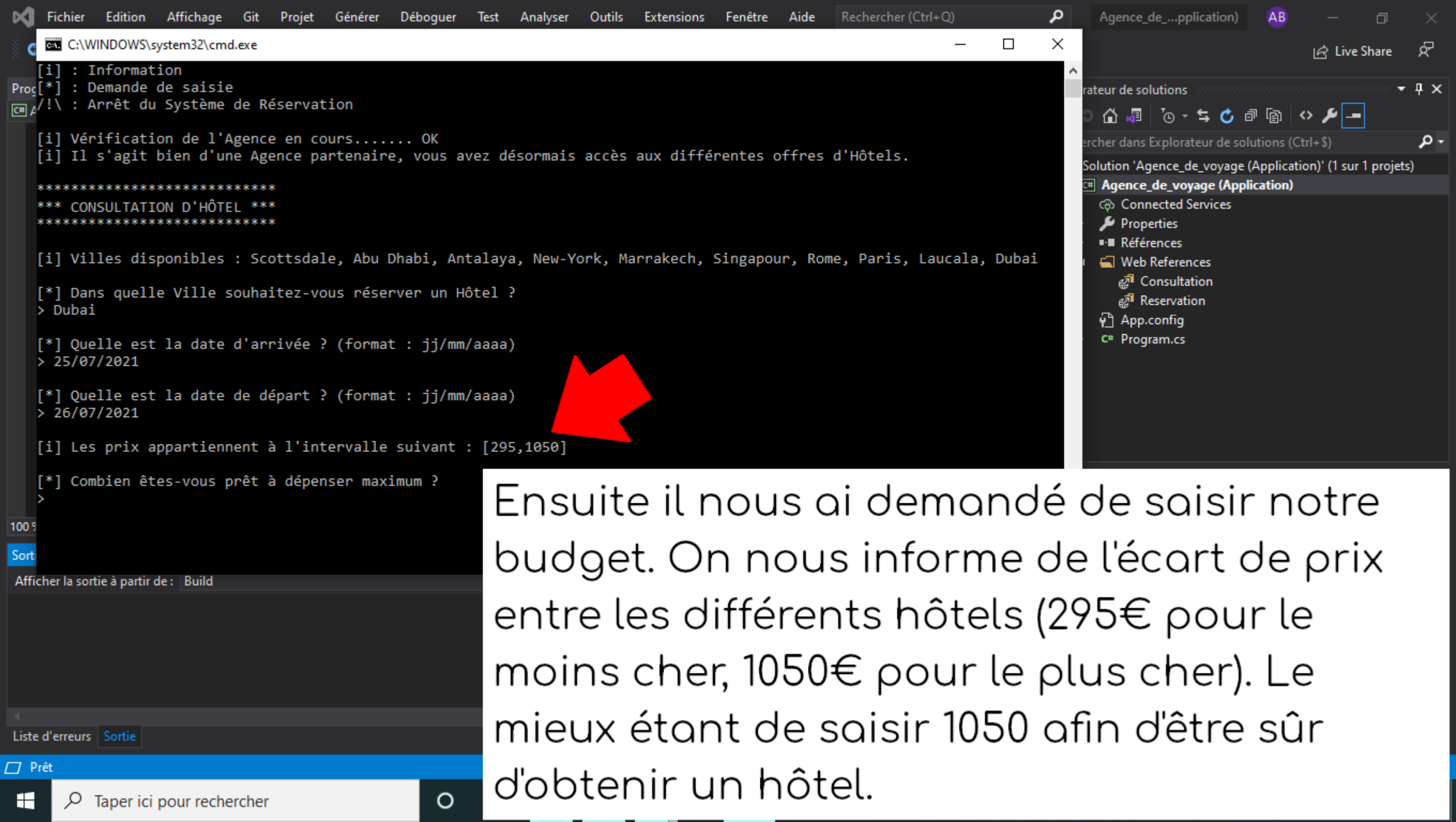


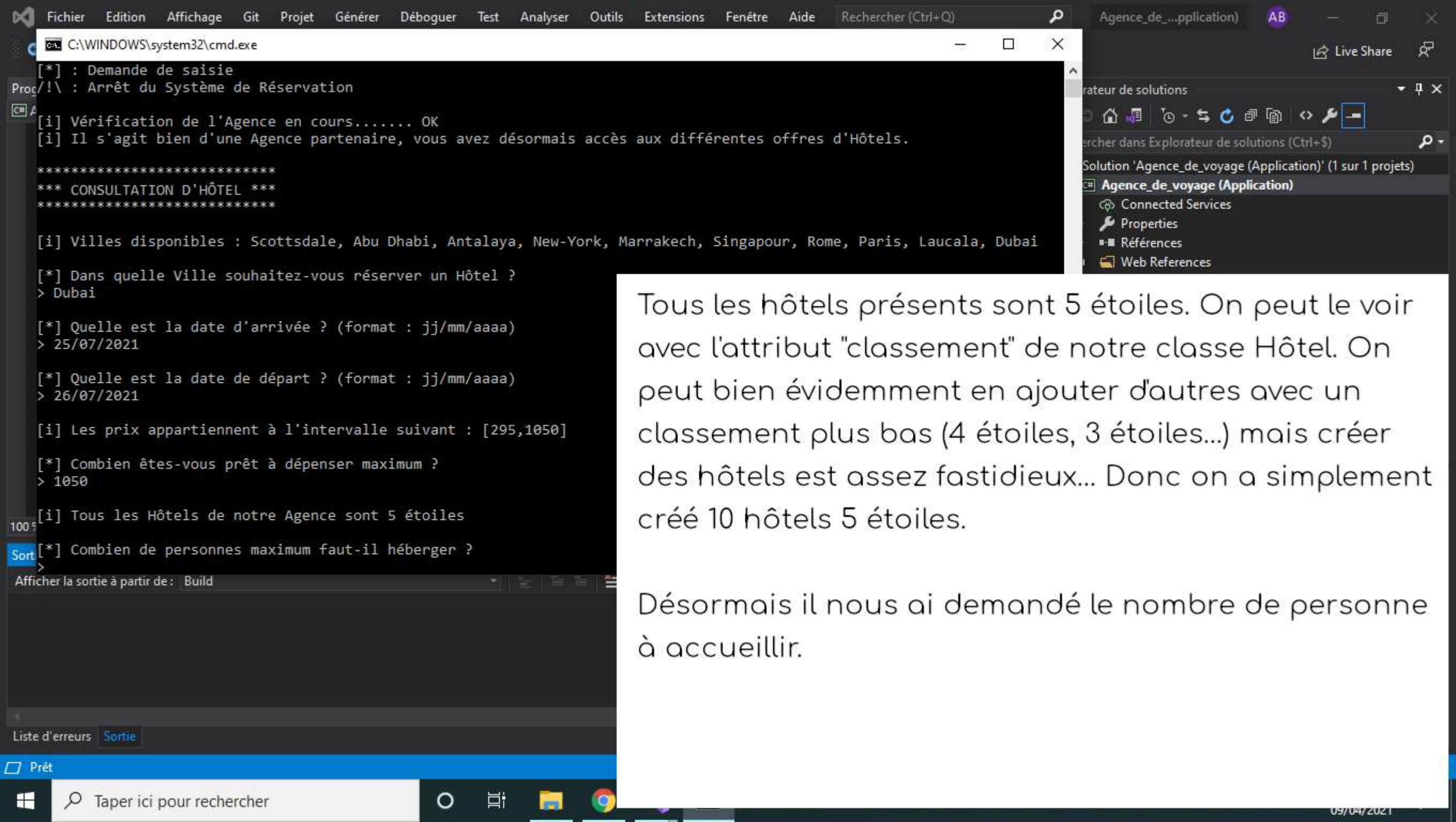


De plus, les données saisies doivent être cohérentes.



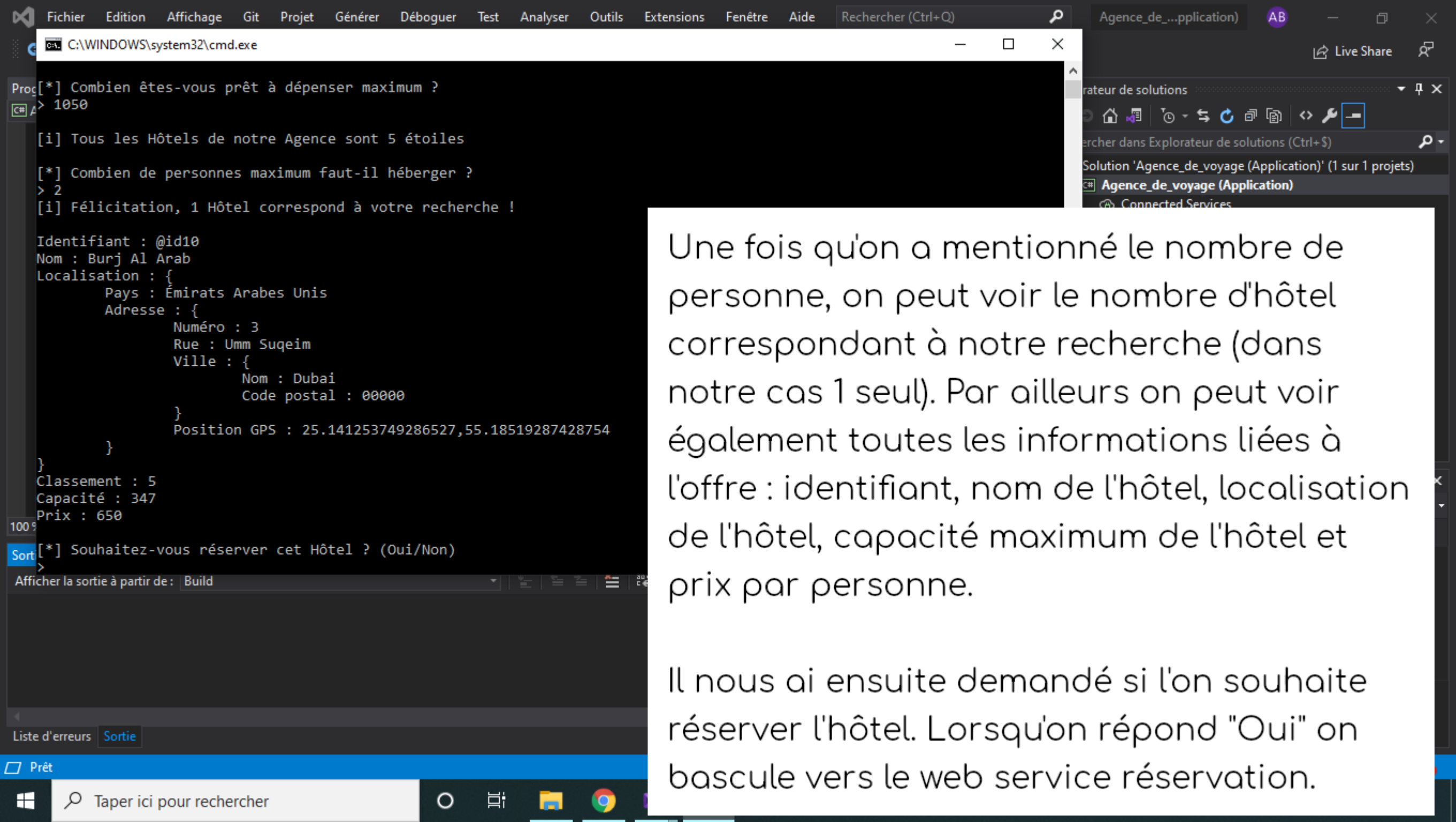






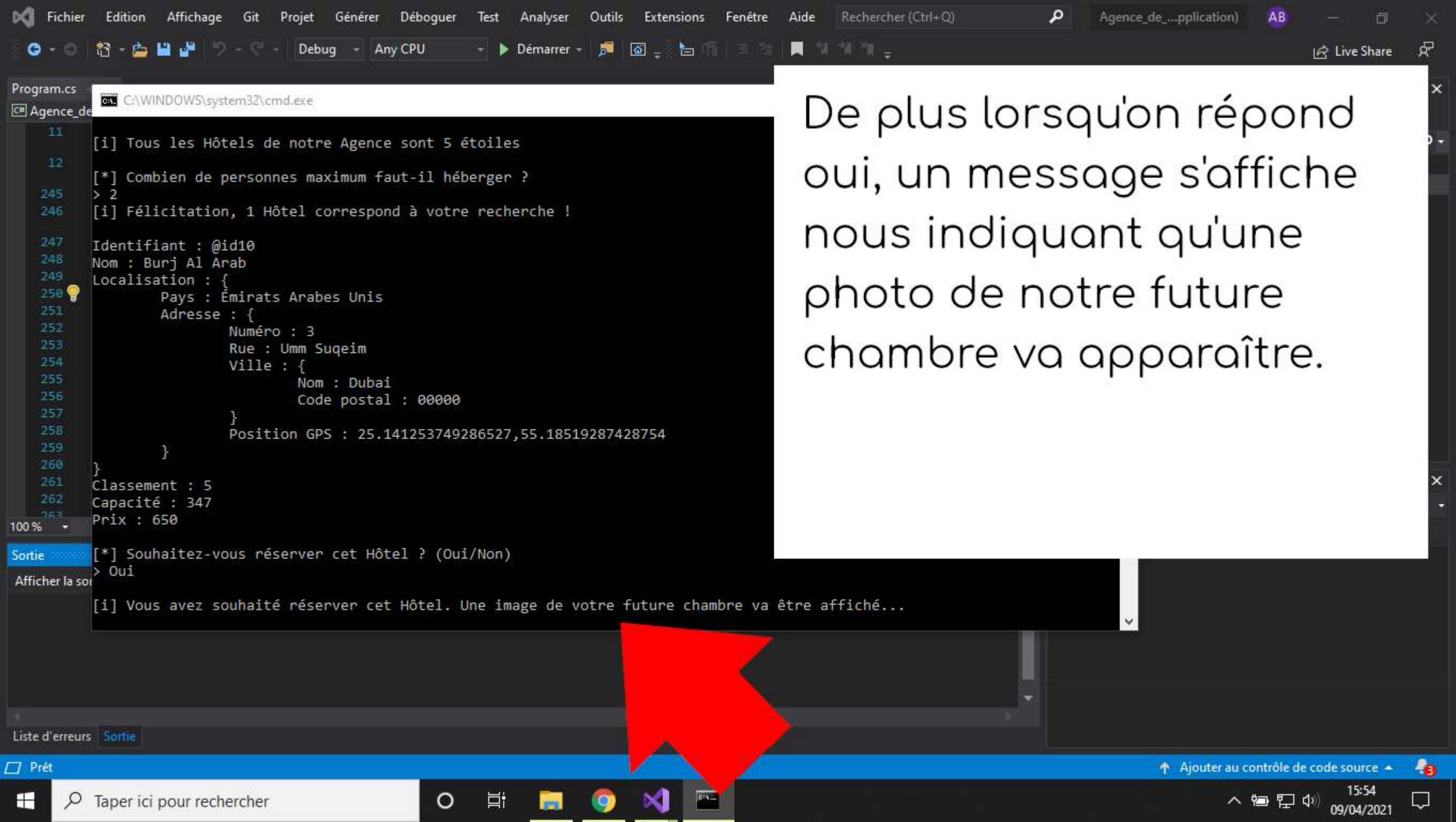
Tous les hôtels présents sont 5 étoiles. On peut le voir avec l'attribut "classement" de notre classe Hôtel. On peut bien évidemment en ajouter d'autres avec un classement plus bas (4 étoiles, 3 étoiles...) mais créer des hôtels est assez fastidieux... Donc on a simplement créé 10 hôtels 5 étoiles.

Désormais il nous a demandé le nombre de personnes à accueillir.



Une fois qu'on a mentionné le nombre de personne, on peut voir le nombre d'hôtel correspondant à notre recherche (dans notre cas 1 seul). Par ailleurs on peut voir également toutes les informations liées à l'offre : identifiant, nom de l'hôtel, localisation de l'hôtel, capacité maximum de l'hôtel et prix par personne.

Il nous a ensuite demandé si l'on souhaite réserver l'hôtel. Lorsqu'on répond "Oui" on bascule vers le web service réservation.



De plus lorsqu'on répond oui, un message s'affiche nous indiquant qu'une photo de notre future chambre va apparaître.

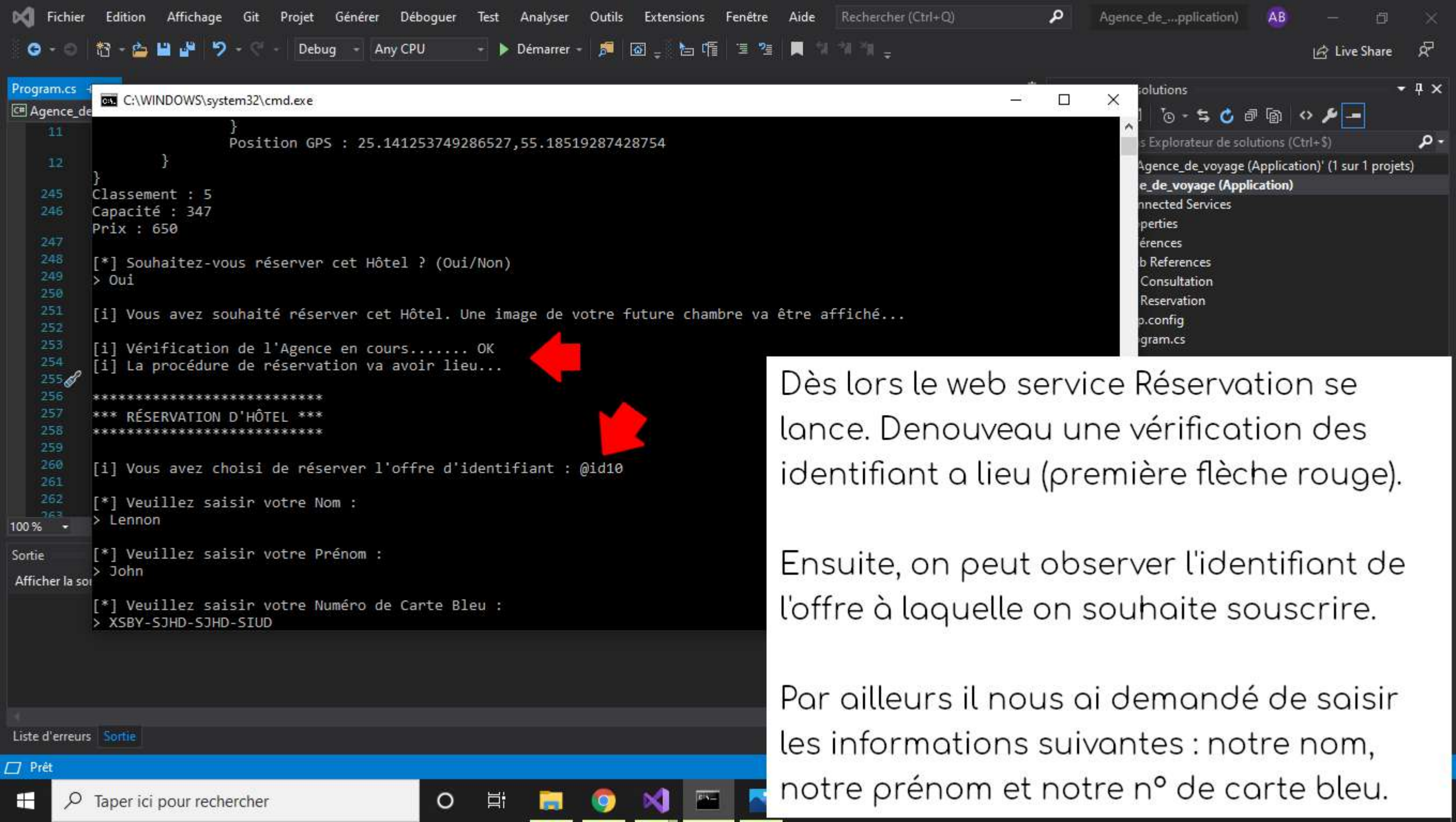
 Voir toutes les photos

Et c'est bien ce qu'il se passe ensuite.



Pour réaliser ceci, le web service contient localement les images. Il envoie à l'application cliente l'image sous la forme d'une string Base64 qui est ensuite convertie en un tableau de byte par le client et puis sauvegardée localement.

Le lancement automatique de l'image se fait via la procédure `System.Diagnostics.Process.Start("chemin vers l'image")`.



Dès lors le web service Réservation se lance. Denouveau une vérification des identifiant a lieu (première flèche rouge).

Ensuite, on peut observer l'identifiant de l'offre à laquelle on souhaite souscrire.

Par ailleurs il nous ai demandé de saisir les informations suivantes : notre nom, notre prénom et notre n° de carte bleu.

FichierEditionAffichageGitProjetGénérerDéboguerTestAnalyserOutilsExtensionsFenêtreAideRechercher (Ctrl+Q)Agence_de_...pplication)AB—

DebugAny CPUDémarrer

Program.cs

Agence_de_voyage (Application)

C:\WINDOWS\system32\cmd.exe

```
226 [i] Vous avez choisi de réserver l'offre d'identifiant : @id10
227
228 [*] Veuillez saisir votre Nom :
229 > Lennon
230
231 [*] Veuillez saisir votre Prénom :
232 > John
233
234 [*] Veuillez saisir votre Numéro de Carte Bleu :
235 > XSBY-SJHD-SJHD-SIUD
236
237 [i] La Réservation a bien eu lieu !
238
239 *****
240 *** RÉCAPITULATIF RÉSERVATION ***
241 *****
242
243 ▶ Nom : Lennon
244 ▶ Prénom : John
245 ▶ Hôtel : Burj Al Arab
246 ▶ Lieu : Émirats Arabes Unis, Dubai
247 ▶ Nombre : 2 personne(s)
248 ▶ Nombre de nuit : 1
249 ▶ Tarif : 1300 euros
250
251 *****
252 [i] Notre Agence vous remercie pour votre confiance, à bientôt !
253 Appuyez sur une touche pour continuer...
```

100 %

Sortie

Afficher la s

Liste d'erreurs

Sortie

Prêt

Taper ici pour rechercher

Enfin lorsque le dernier champ est rempli, on obtient une référence de la réservation et on affiche les informations de cette réservation.

Le nombre de nuit est déduit au moment de la saisie date arrivée/départ.

Le montant de la réservation se fait via le calcul suivant :

Tarif Hôtel x Nombre de Personne x Nombre de nuit

Dans notre cas :

Tarif Hôtel = 650
Nombre personne = 2
Nombre de nuit = 1

On a donc $650 \times 2 \times 1 = 1300\text{€}$