**BrokerCompanySystem**

**Avaliable from**: Sunday, 25 October 2020, 10:00  
**Due date**: Sunday, 25 October 2020, 16:00  
**Maximum number of files**: 9  
**Type of work**: Individual work

**Broker Company System**

**Преглед на задачата**

Имате за задача да помогнете на един приятел с курсовата му работа по програмиране. Задачата му е да направи система за регистриране на брокерски компании и да създаде архив с информация за каталозите от сгради на всяка фирма и да проследи работата на служителите си – брокерите.

**Problem 1.       Структура**

**Building**

Сградата е **базов** **клас** за всички видове сгради и **не** **трябва** да може да се **инстанцира**.

**Data**

Всяка сграда има име, град, звезди, сума за наем и статус.

**Name** – символен низ

**City** – символен низ

**Stars** – цяло число

**RentAmount** – реално число

**IsAvailable**– булева стойност

* Името на сградата **не** **трябва** да бъдат **null** или **празен** **стринг.**

**Съобщение: Building name** **must not be null or empty!**

* Града на сградата **не** **трябва** да бъдат **null** или **празен** **стринг.**

**Съобщение:**City **must not be null or empty!**

* Звездите на сградата **не може**да бъдат **отрицателни**или **повече от 5.**

**Съобщение:**Stars **cannot be less than 0 or above 5!**

* Наема на сградата **не може**да бъдат **отрицателен**или равен на **0.**

**Съобщение:**Rent amount cannot be less or equal to 0!

* При подаване на **невалидна** стойност хвърлете изключение – **ArgumentException с конкретното съобщение**.

**Constructor**

Това са параметрите за конструктора:

string name, string city, int stars, double rentAmount

**Behavior**

**override string ToString()**

**\*\*\*\*Building: {Name} <Stars>**

**\*\*\*\*Location: {City}**

**\*\*\*\*RentAmount: {RentAmount}**

**\*\*\*\*Is Available: {IsAvailable}**

**Child Classes**

**Hotel**

Името трябва да завършва на **Hotel**.

**Example**: Green Park Hotel

При невалидно име хвърлете **ArgumentException**.

**Съобщение: Name of hotel buildings should end on Hotel!**

**Residence**

Името трябва да завършва на **Residence**.

**Example**: Green Park Residence

При невалидно име хвърлете **ArgumentException**.

**Съобщение: Name of residence buildings should end on Residence!**

**Business**

Името трябва да завършва на **Business**.

**Example**: Green Park Business

При невалидно име хвърлете **ArgumentException**.

**Съобщение: Name of business buildings should end on Business!**

**Broker**

**Data**

Всеки брокер има име, години, град, бонус и **поле**със сгради, които той е отдал:

Name – символен низ

Age – цяло число

**City**– символен низ

Bonus – реално число

* Името на брокера **не** **трябва** да бъдат **null** или **празен** **стринг.**

**Съобщение: Broker name** **must not be null or empty!**

* Годините на брокера не може да са по – малко от 16 и повече от 70.

**Съобщение: Broker's age cannot be less than 16 or above 70!**

* Града на брокера **не** **трябва** да бъдат **null** или **празен** **стринг.**

**Съобщение:**City **must not be null or empty!**

* При подаване на **невалидна** стойност хвърлете изключение – **ArgumentException с конкретното съобщение**.

**Constructor**

Това са параметрите за конструктора:

string name, int age, string city

**Behavior**

**double ReceiveBonus(Building building)**

Трябва да се изчисли и добави бонуса на брокера отдал сградата и трябва да се добави сградата в списъка му с отдадени сгради. Върнете калкулирания текущ бонус изчислен по следната формула:

**{Building rent amount} \* 2 \* {building stars} / 100**

**override string ToString()**

**\*\*\*\*Broker: {Name} <{Age}>**

**\*\*\*\*Location: {City}**

**\*\*\*\*Bonus: {Bonus}**

**\*\*\*\*\*\* {building1.Name}**

**\*\*\*\*\*\* {building2.Name}**

**…**

**\*\*\*\*\*\* {buildingN.Name}**

**Company**

**Data**

Всяка компания има име, **поле** със сгради и **поле** с брокери:

**Name** – символен низ

* Името на компанията **не** **трябва** да бъдат **null** или **празен** **стринг.**

**Съобщение: Company name** **must not be null or empty!**

**Constructor**

string name

**Behavior**

**void AddBroker(Broker broker)**

Трябва да добави брокера към списъка със служители на компанията. В случай, че брокер с такова име вече фигурира просто не го добавяйте.

**void AddBuilding(Building building)**

Трябва да добави сградата към списъка със сгради на компанията. В случай, че сграда с такова име вече фигурира просто не я добавяйте.

**Broker GetBrokerByName(string name)**

Връща брокер с предоставеното име или **NULL.**

**Building GetBuildingByName(string name)**

Връща сграда с предоставеното име или **NULL.**

**override string ToString()**

**Company: {company name}**

**##Brokers: {count of brokers}**

**{ broker1.Tostring() }**

**{ broker2.Tostring() }**

**...**

**{ brokerN.Tostring() }**

**##Buildings – {count of buildings}**

**{ building1.Tostring() }**

**{ building2.Tostring() }**

**...**

**{ buildingN.Tostring() }**

**Problem 2.       Бизнес логика**

**The Controller Class**

Бизнес логиката на програмата трябва да бъде концентрирана около няколко команди. Внедрете клас, наречен **CompanyController**, който ще притежава главната функционалност, представена от тези публични методи:

|  |
| --- |
| CompanyController.cs |
| public string CreateCompany(List<string> args)  {      //TODO: Add some logic here …  }    public string RegisterBuilding(List<string> args)  {      //TODO: Add some logic here …  }    public string RegisterBroker(List<string> args)  {      //TODO: Add some logic here …  }    public string RentBuilding(List<string> args)  {      //TODO: Add some logic here …  }    public string CompanyInfo(List<string> args)  {      //TODO: Add some logic here …  }    public string Shutdown()  {      //TODO: Add some logic here …  } |

**ЗАБЕЛЕЖКА: Не трябва да променяте нищо по методите. Трябва да имплементирате логиката на самите методи. Не прихвайщайте никакви изключения!**

**Команди**

Има няколко команди, които контролират бизнес логиката на приложението и трябва да ги имплементирате.

Те са посочени по-долу.

**CreateCompany Команда**

Създава брокерска компания и я регистрира в системата.

**Не всички данни ще бъдат валидни!!!**

**Параметри**

* name – символен низ

**RegisterBuilding Команда**

Създава сграда и я добавя в каталога на дадена компания.

**Забележка: Типа сграда винаги ще е валиден!!!**

**Не всички данни ще бъдат валидни!!!**

**Параметри**

* **type**– символен низ
* name – символен низ
* **city**– символен низ
* stars – цяло число
* **rentAmount**  – реално число
* **companyName**– символен низ

**RegisterBroker Команда**

Създава брокер и го добавя в служителите на дадена компания.

**Не всички данни ще бъдат валидни!!!**

**Параметри**

* name – символен низ
* age – цяло число
* **city**– символен низ
* **companyName**– символен низ

**RentBuilding Команда**

При тази команда една от сградите ще бъде наета. Сградата трябва да се маркира като заета и брокера трябва да получи своя бонус.

**Не всички данни ще бъдат валидни!!!**

**Параметри**

* companyName – символен низ
* buildingName – символен низ
* brokerName – символен низ

**CompanyInfo Команда**

При тази команда се връща състоянието на компанията във формат описан долу в секцията за вход и изход.

**Параметри**

* **companyName** – символен низ

**Shutdown Команда**

Тази команда прекратява изпълнението на програмата и връща всички компании, подредени по име в нарастващ ред.

**Problem 3.       Вход / Изход**

**Вход**

* Четете редове с различни команди, докато не получите команда за приключване на програмата.

По-долу можете да видите формата, в който всяка команда ще бъде дадена във входа:

* CreateCompany {name}
* RegisterBuilding {type} {name} {city} {stars} {rentAmount} {companyName}
* RegisterBroker {name} {age} {city} {companyName}
* RentBuilding {companyName} {buildingName} {brokerName}
* **CompanyInfo {companyName}**
  + Shutdown

**Изход**

По – долу може да видите кой изход трябва да бъде предоставен от командите.

**CreateCompany Команда**

При успешно регистриране върнете:

**Съобщение**: **Company {company name} was successfully registerd in the system!**

В случай, че компания с това име вече съществува, просто я пропуснете и върнете:

**Съобщение**: **Company {company name} is already registered!**

**RegisterBuilding Команда**

При успешно регистриране върнете:

**Съобщение**: **Building {building name} was successfully registerd in {company name} catalog!**

В случай, че компанията не съществува върнете:

**Съобщение**: **Invalid Company: {company name}. Cannot find it in system!**

В случай, че сграда със същото име фигурира вече в каталога на компанията просто я пропуснете и върнете:

**Съобщение**: **Building {building name} is already registered in {company name}!**

**RegisterBroker**

При успешно регистриране върнете:

**Съобщение**: **Broker {broker name} was successfully hired in {company name}!**

В случай, че компанията не съществува върнете:

**Съобщение**: **Invalid Company: {company name}. Cannot find it in system!**

В случай, че служител със същото име фигурира вече просто го пропуснете и върнете:

**Съобщение**: **Broker {broker name} is already part of {company name}!**

**RentBuilding**

При успешно отдаване на сграда върнете:

**Съобщение**: **Successfully rented {building name}!**

**Broker {broker name} earned {bonus amount}!**

В случай, че компанията не съществува върнете:

**Съобщение**: **Invalid Company: {company name}. Cannot find it in system!**

В случай, чe сградата в компанията не съществува върнете:

**Съобщение**: **Invalid Building: {building name}. Cannot find it in {company name} catalog!**

В случай, чe сградата вече е наета:

**Съобщение**: **Building: {buildingName} cannot be rented!**

В случай, че брокера в компанията не съществува върнете:

**Съобщение**: **Invalid Broker: {broker name}. Cannot find employee in {company name}!**

**CompanyInfo**

При успешно намиране на компания върнете информация за нея във формата:

**Company: {company name}**

**##Brokers: {count of brokers}**

**{ broker1.ToString() }**

**{ broker2.ToString() }**

**...**

**{ brokerN.ToString() }**

**##Buildings: {count of buildings}**

**{ building1.ToString() }**

**{ building2.ToString() }**

**...**

**{ buildingN.ToString() }**

В случай, че компанията не съществува върнете:

**Съобщение**: **Invalid Company: {company name}. Cannot find it in system!**

**Shutdown**

След завършване на командата трябва да върнете информация за всички компании в системата, подредени по име на компанията във формата, описан в **CompanyInfo командата. Преди това добавете следния ред:**

**Companies: {company count}**

**Ограничения**

* Имената на компаниите, сградите и брокерите ще бъдат символни низове, които може да съдържат всеки ASCII символ, с изключение на звезда(\*).
* Винаги ще получавате команда за приключване на програмата.
* Входните данни ще бъдат валидни от страна на типове данни.

**Примери**

|  |  |
| --- | --- |
| **Input** | **Output** |
| CreateCompany\*Deager  RegisterBroker\*Sade\*35\*Plovdiv\*Deager  RegisterBuilding\*Hotel\*Sun Hotel\*Sofia\*4\*50000\*Deager  RegisterBroker\*Roy\*18\*Sofia\*Deager  CompanyInfo\*Deager  RegisterBuilding\*Business\*Heaven Business\*Plovdiv\*5\*80000\*Deager  RegisterBuilding\*Residence\*Hell Residence\*Plovdiv\*2\*20000\*Deager  RentBuilding\*Deager\*Heaven Business\*Sade  RentBuilding\*Deager\*Hell Residence\*Sade  RentBuilding\*Deager\*Sun Hotel\*Roy  Shutdown | Company Deager was successfully registerd in the system!  Broker Sade was successfully hired in Deager!  Building Sun Hotel was successfully registerd in Deager catalog!  Broker Roy was successfully hired in Deager!  Company: Deager  ##Brokers: 2  \*\*\*\*Broker: Sade <35>  \*\*\*\*Location: Plovdiv  \*\*\*\*Bonus: 0  \*\*\*\*Broker: Roy <18>  \*\*\*\*Location: Sofia  \*\*\*\*Bonus: 0  ##Buildings: 1  \*\*\*\*Building: Sun Hotel <4>  \*\*\*\*Location: Sofia  \*\*\*\*RentAmount: 50000  \*\*\*\*Is Available: True  Building Heaven Business was successfully registerd in Deager catalog!  Building Hell Residence was successfully registerd in Deager catalog!  Successfully rented Heaven Business!  Broker Sade earned 8000!  Successfully rented Hell Residence!  Broker Sade earned 800!  Successfully rented Sun Hotel!  Broker Roy earned 4000!  Companies: 1  Company: Deager  ##Brokers: 2  \*\*\*\*Broker: Sade <35>  \*\*\*\*Location: Plovdiv  \*\*\*\*Bonus: 8800  \*\*\*\*\*\* Heaven Business  \*\*\*\*\*\* Hell Residence  \*\*\*\*Broker: Roy <18>  \*\*\*\*Location: Sofia  \*\*\*\*Bonus: 4000  \*\*\*\*\*\* Sun Hotel  ##Buildings: 3  \*\*\*\*Building: Sun Hotel <4>  \*\*\*\*Location: Sofia  \*\*\*\*RentAmount: 50000  \*\*\*\*Is Available: False  \*\*\*\*Building: Heaven Business <5>  \*\*\*\*Location: Plovdiv  \*\*\*\*RentAmount: 80000  \*\*\*\*Is Available: False  \*\*\*\*Building: Hell Residence <2>  \*\*\*\*Location: Plovdiv  \*\*\*\*RentAmount: 20000  \*\*\*\*Is Available: False |
| CreateCompany\*Deager  CreateCompany\*  RegisterBroker\*Sade\*35\*Plovdiv\*Deager  RegisterBroker\*John\*27\*Plovdiv\*InvalidCompany  RegisterBuilding\*Hotel\*Sun Business\*Sofia\*4\*50000\*Deager  RegisterBroker\*\*20\*Varna\*Deager  RegisterBuilding\*Business\*Sun Business\*Sofia\*4\*50000\*Deager  RegisterBroker\*Roy\*10\*Ruse\*Deager  RegisterBuilding\*Business\*Heaven Business\*Plovdiv\*5\*-8000\*Deager  RentBuilding\*MissingCompanyName\*Sun Business\*Roy  Shutdown | Company Deager was successfully registerd in the system!  Company name must not be null or empty!  Broker Sade was successfully hired in Deager!  Invalid Company: InvalidCompany. Cannot find it in system!  Name of hotel buildings should end on Hotel!  Broker name must not be null or empty!  Building Sun Business was successfully registerd in Deager catalog!  Broker's age cannot be less than 16 or above 70!  Rent amount cannot be less or equal to 0!  Invalid Company: MissingCompanyName. Cannot find it in system!  Companies: 1  Company: Deager  ##Brokers: 1  \*\*\*\*Broker: Sade <35>  \*\*\*\*Location: Plovdiv  \*\*\*\*Bonus: 0  ##Buildings: 1  \*\*\*\*Building: Sun Business <4>  \*\*\*\*Location: Sofia  \*\*\*\*RentAmount: 50000  \*\*\*\*Is Available: True |

**Problem 4.       Точкуване**

Всяка различна задача Ви дава точки:

* Структура - 30 точки
* Бизнес логика - 50 точки
* Вход/Изход - 20 точки