Create a housing application for a property manager.

1) Create a **class** named **Home**.
   a) Include data characteristics as follows:
      i) **Address** as a string;
      ii) **Year built** (read-only, must be between 1800 and 2018);
      iii) **Price** (positive decimal);
      iv) (make sure you have public properties that allow you to get these values, but these values should not be set outside of this class)
      v) Throw an **ArgumentOutOfRangeException** if the value is invalid.
   b) Include a **protected constructor** that initializes all these data characteristics in the same order as above.
   c) Include the following members in the **Home** as follows:
      i) **Total Cost** as a **virtual** property that returns the price in **decimal**.
      ii) **Get Rate** as an **virtual** method that takes the **number of periods** in integer and **returns** a **decimal**. This method, by default, throws a **NotImplementedException**.
   d) **Override** the **ToSting** method to include the **address** and **year built**, separated by a single space.
2) Create a class named **Condo** that extends the **Home** class.
   a) Include data characteristics in addition to the **Home**'s as follows:
      i) **Unit Number** (e.g. Apartment Number) in String,
      ii) **Fee** (must be a positive value) that returns a **decimal**;
      iii) (make sure you have **public** properties that allow you to get these values, but these values should not be set outside the class)
      iv) Throw an **ArgumentOutOfRangeException** if the value is invalid.
   b) Include the **IsRental** property that allows you to **set** and **get** publicly.
   c) The constructor should initialize all the data members in the **base class** plus all the data members in this **class** in the exact same order. Also, have another constructor (or the same constructor) that defaults the **is rental** property to **false**.
   d) **Override** the **total cos**t property to return the **base's Total Cost** plus the **fee**.
   e) The **GetRate override** method will return the **Total Cost** divided by the **number of periods**.
   f) The **Get Monthly Rate** method will do the following:
      i) If the **is rental** property is **false**, then throw **InvalidOperationException**; otherwise,
      ii) return **150%** of the rate in 360 months (use the **Get Rate** method and **360** as the number of periods).
   g) The **ToString** override method returns base's **to string** method with the **unit number**, separated by a space. The following costs will follow by a space after the string:

    i) Also, it will include the **`total cost`** if the condo is not a rental;
    ii) Otherwise if it's a rental, the return string will include the monthly rate from the **`get monthly rate`** method instead.
    iii) (make sure that these prices are formatted with string formatter's "**`C`**")
3) Create a derived class named **`SingleFamily`** that extends the **`Home`** class.
    a) There will be no new stuff added to this class, but make sure you have implemented everything, such as constructors and methods.
    b) For the **`Get Rate method`**, simply **`return`** the **`total cost`** divided by the **`number of periods.`**
    c) The **`ToString()`** method also includes the **`Total Price`** of the home. (make sure that the price is formatted with string formatter's "**`C`**")
4) Test your **`classes.`**