**Part II Bike Shop Requirements:**

NOTE: When asked to provide professional content, part of your grade will be based on how professional the content is. Create/use GitHub repository/ ASP.NET Project both called BikeShop. Version both daily. No flash drives.

1. If you have not done so already, complete both the short and long *professional* descriptions for each product. Again, they should be “real”. Update the database records accordingly. **– 10 points**
2. Correctly complete all of the Bike Shop code we have done so far; including the Home page and Store page material from Part I and the Shopping Cart page (and any other) material from Part II. **– 50 points**
3. Continue to add Bootstrap to the project where necessary to keep it looking professional (e.g. in the checkout forms). – **10 points**
4. Using the material in the article: **Tutorial: Enhance data validation for EF Database First with ASP.NET MVC app** both emailed to you earlier this week and available at URL: <https://docs.microsoft.com/en-us/aspnet/mvc/overview/getting-started/database-first-development/enhancing-data-validation> from the **Add metadata classes** of this document, actually go and look at the original schema, determine where it is appropriate to add metadata classes, and do it. – **15 points**
5. Complete your Home, About and Contact pages. Again, the Contact form should include a professional Bootstrap form, and the Home and About page should look like those of a *real bike shop*. **–5 points each, 15 points total**.
6. As explained in class, change the Expire Year for the Credit Card from a hard coded 2019, 2020, …, 2030 to a programmatically done current year + 11 years. The result (at least until 12/31/2019) should look the same as it does currently. But as of 1/1/2020, if a person ran the program it would show the years 2020, 2021, …, 2031. **– 10 points**
7. This we did **not** go over in class. Do the following:
   1. Add an additional entry to a person’s bill. This will be the taxable total (assume all purchases are taxed at 10%). Make sure that as the user adds/removes products from his/her cart that both the subtotal (total before tax), taxable total, and final total (subtotal + taxable total) are both shown and updated accordingly. **– 10 points**
   2. Once the final purchase has been made, update the OrderComplete page to provide the following:
8. Give the user and order confirmation number. This should be a combination of the date and time (to the hundredth of a second). So, if a purchase was made today, 10/30/2019 at 6:07 a.m. and 45/100 of a second, the confirmation number would be 10302019060745. Be sure to “pad” with 0’s as necessary and shown in the example above. Provide this number on the receipt created below **– 10 points**
9. Provide the user with the equivalent of a receipt either on the OrderComplete page or on an additional page (your choice). Design the receipt yourself but make sure it has the *look-and-feel* of the rest of the site. **– 20 points**