

1.
 - [].class : Array
 - 3/2 : 1
 - 3.0 / 2.0 : 1.5
 - [].nil? : false
 - def h; "Hello world"; end: nil
2. I like it that everything is an object. Since everything is an object, every bit of information and code can be given their own properties and actions, and can be accessed the same way, it just makes things more uniformed and easier.
3. It would be worth it in some cases, for example for a startup BDD can help them see better of their necessary features and where the revenue would be since it represents more of the features than TDD. At the end of the day, most products are going to go to the customers and it is important to have way to let them understand the products.
4. It is a bit different to what I'm used to, but I do see where it can be useful and helpful. I find it interesting and like it so far.
5. Advantage is that it would be more descriptive and easier for human reader to understand the behavior. The drawback is that there would need to be definitions in steps to tell the machine what it means.