# And Now C++: CarRental, answer template

1. Multiple include protection is needed in C and C++ because:  
   Zodat er geen dubbele includes ontstaan, dus dat er in een niet file meerdere keren dezelfde code staat.  
     
   Used source(s): https://en.wikipedia.org/wiki/Include\_guard
2. In C++ you declare a variable or method public or private by:  
   Je kan met public: aangegeven dat de variabelen en/of methodes erna public moeten zijn. Als ze private moeten zijn kun je deze voor de public: zetten of na een private:.  
     
   Used source(s): http://www.cplusplus.com/doc/tutorial/classes/
3. A decent replacement for Decimal in C++ is:  
   double  
   Because: Het wordt ook gebruikt om decimalen getallen in op te slaan, en is qua grootte(64bit) het grootst in C++ ookal is de decimal in c# nog groter(128 bit).  
     
   Used source(s): https://stackoverflow.com/questions/618535/difference-between-decimal-float-and-double-in-net
4. What is a typical way to implement properties in C++:  
   Een private field aanmaken en dan in de public de getter en setter als functie apart aanmaken.  
     
   Used source(s): https://www.codeproject.com/Articles/118921/C-Properties
5. virtual means:  
   Een functie in de base class met een implementatie die je kan overschrijven in afgeleide klassen.  
     
   Used source(s): https://stackoverflow.com/questions/2391679/why-do-we-need-virtual-functions-in-c
6. In C++ you make a class abstract by:  
   Het hebben van een pure virtual function in een klasse.  
     
   Used source(s): https://docs.microsoft.com/en-us/cpp/cpp/abstract-classes-cpp?view=vs-2019
7. To override inherited behaviour in C++ you ***must*** *(please do note that the override keyword in C++ is optional)*: In de base klasse en afgeleide klasse een methode met dezelfde naam en verschillende implementaties.  
     
   Used source(s): https://www.programiz.com/cpp-programming/function-overriding
8. Making a method const in C++ is considered "programming wisely" because:  
   Als je const gebruikt bij een methode krijg je een compiler error als je data in de klasse aanpast, wat niet de bedoeling is bij een const method.  
     
   Used source(s): https://stackoverflow.com/questions/3141087/what-is-meant-with-const-at-end-of-function-declaration
9. In C++ you make a method const by:  
   Keyword const achter de declaratie van de methode zetten.  
     
   Used source(s): https://stackoverflow.com/questions/3141087/what-is-meant-with-const-at-end-of-function-declaration
10. The following methods should be made const:  
    Alle methodes waar er geen data aangepast wordt.  
      
    Used source(s): https://stackoverflow.com/questions/3141087/what-is-meant-with-const-at-end-of-function-declaration
11. The given implementation of GetKilometers with prototype int GetKilometers() doesn't work because: De method weet nu niet waar hij de kilometers moet zoeken, er moet Car:: voor de naam van de functie zodat je hier de properties van kan gebruiken.  
      
      
    Used source(s): https://stackoverflow.com/questions/28283598/where-should-i-implement-my-class-method
12. You can change a parameter from value type to reference type in C and C++ by:  
    Door een & teken achter het type van de variabele te zetten. Bijvoorbeeld int i = 17; is een value type, int& r = i, r gedraagt zich als een reference type naar i.  
      
    Used source(s): https://www.tutorialspoint.com/cplusplus/cpp\_references.htm
13. Do any of the Car constructor parameters have to be reference type (and why)?  
    Nee, de parameters kunnen normaal aan de fields en properties gesteld worden.  
      
    Used source(s): Niet gevonden
14. How do exceptions work in C++? Can you name the differences with C#?  
    Bijna hetzelfde als in C#, met een try en catch blok, alleen de parameters van de catch zijn anders en kunnen ook int of char zijn bijvoorbeeld.  
      
    Used source(s): http://www.cplusplus.com/doc/tutorial/exceptions/
15. A good C++ candidate for ArgumentOutOfRangeException and ArgumentNullException would be: Voor de eerste is er een eigen out\_of\_range exception uit de “stdexcept” library. Voor de null pointer exception geeft het geen aparte exception, je kunt well catchen met catch(…) maar dat is niet een ‘best practice’, je kunt wel van te voren een check doen of een object == 0 is.  
      
    Used source(s): <https://stackoverflow.com/questions/2709719/throwing-out-of-range-exception-in-c> en https://stackoverflow.com/questions/5271969/c-exception-for-null-pointers
16. You inherit a class in C++ by:  
    Bijna hetzelfde als C#, dus achter de naam van de klasse : met de naam van de base klasse alleen moet voor de naam nog een access specifier voor de base klasse.  
      
    Used source(s): https://www.geeksforgeeks.org/inheritance-in-c/
17. In C++ you cannot call the base constructor by calling base, because:  
    In plaats van base moet je de naam van de klasse die base representeerd typen.  
      
    Used source(s): https://stackoverflow.com/questions/120876/what-are-the-rules-for-calling-the-superclass-constructor
18. **Optional**: the C++ list type is not comparable to the C# list type because:  
    Een C++ list is overeenkomstig met en C# linkedlist, en een C# list is overeenkomstig met een C++ vector.  
      
    Used source(s): https://stackoverflow.com/questions/17013906/is-a-c-sharp-list-like-a-c-list
19. You cannot declare vector<Car> cars; because Car is abstract. This is solved by:  
    Je moet in plaats daarvan pointers naar car gebruiken.  
      
    Used source(s): https://stackoverflow.com/questions/2160920/why-cant-we-declare-a-stdvectorabstractclass
20. cin.ignore() is needed because:  
    Het zorgt ervoor dat er geen characters meer na de cin command voor problemen kunnen zorgen. Alles wat erna komt wordt genegeerd als het ware.  
      
    Used source(s): https://www.quora.com/Can-you-explain-cin-ignore-and-cin-clear-for-C-When-is-it-appropriate-to-use-these-How-do-they-affect-the-next-cin
21. Shared libraries are:  
    Libraries die bij het starten van een programma geladen worden en gebruikt kunnen worden, waarmee je voorgeprogrammeerde functies en classes etc. kunt gebruiken. Shared libraries werken dus als een losse file en moeten beschikbaar zijn bij het compileren en als de applicatie start.  
      
    Used source(s): https://www.tecmint.com/understanding-shared-libraries-in-linux/
22. I would use shared libraries in these cases:  
    Bijvoorbeeld als je een wiskundige functie nodig hebt hoef je deze niet zelf te schrijven maar kan je deze uit de Math library halen.  
      
    Used source(s): https://www.tecmint.com/understanding-shared-libraries-in-linux/
23. Static libraries are:  
    Een library die in het programma zelf komt als één applicatie, het wordt automatisch in het programma gelinkt wanneer het gebouwd wordt. De library moet beschikbaar zijn bij het compileren en niet bij het starten van de applicatie.  
      
    Used source(s): https://stackoverflow.com/questions/9688200/difference-between-shared-objects-so-static-libraries-a-and-dlls-so
24. I would use static libraries in these cases:  
    Als je zekerheid nodig hebt om te verzekeren dat de binary niet veel externe afhankelijkheden heeft die lastig kunnen zijn.  
      
    Used source(s): https://stackoverflow.com/questions/2649334/difference-between-static-and-shared-libraries