1. Code was converted from Java to C#. This meant translating a lot of commands that are built into Java into equivalents.
2. Renamed classes. The C# language has capitalized functions as standard. For example:

crushingBlow(DungeonCharacter opponent)

becomes

CrushingBlow(DungeonCharacter opponent)

1. Replaced Keyboard.java with Console.ReadLine();
2. Added overrides to DC\_Hero subclass methods so that C# can use them

public void battleChoices(DungeonCharacter opponent)

becomes

public override void BattleChoices(DC opponent)

1. Removed theHero declaration in Dungeon.ChooseHero(). It was unused.
2. changed public hero fields to private where it was necessary

public void ReadName()

becomes

private void ReadName()

1. Added interface DC for DungeonCharacter to implement. All calls to DungeonCharacter now call DC.

DungeonCharacter theHero = chooseHero();

becomes

DC theHero = chooseHero();

1. Added interface DM that can manage menus. Will be useful for future extension
2. Removed all switch-cases outside of DungeonMenu since DungeonMenu automatically determines if your choice is valid.

switch(choice)

{

case 1: return new Warrior();

case 2: return new Sorceress();

case 3: return new Thief();

default: System.out.println("invalid choice, returning Thief");

}

becomes

while (true)

{

Console.WriteLine("\nChoose an option: ");

String choice = Console.ReadLine();

if(int.TryParse(choice, out c))

{

if (c < options.Length + 1 && c >0)

break;

}

Console.WriteLine("Invalid choice!\n");

DisplayMenu();

}

1. Implemented new class CharacterList that centralizes how heroes and monsters are generated on a single class.
2. DC declarations from outside the do statement removed. They now declare within the initial do.

DC theHero;

becomes

DC theHero = chooseHero();