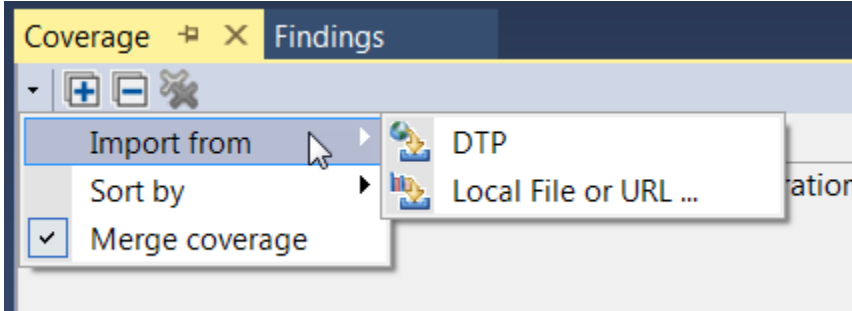


Viewing Coverage

You can import and view coverage results from a local execution XML file or from DTP. The Coverage view shows the percentage of coverage, as well as how many lines of code are covered with tests of the available lines in the file.

1. Choose **Parasoft> Show View> Coverage** from the main menu.
2. Click the coverage drop down menu.



3. Choose your options for viewing coverage information:

Import from: see [Importing Coverage](#) for more information about importing coverage.

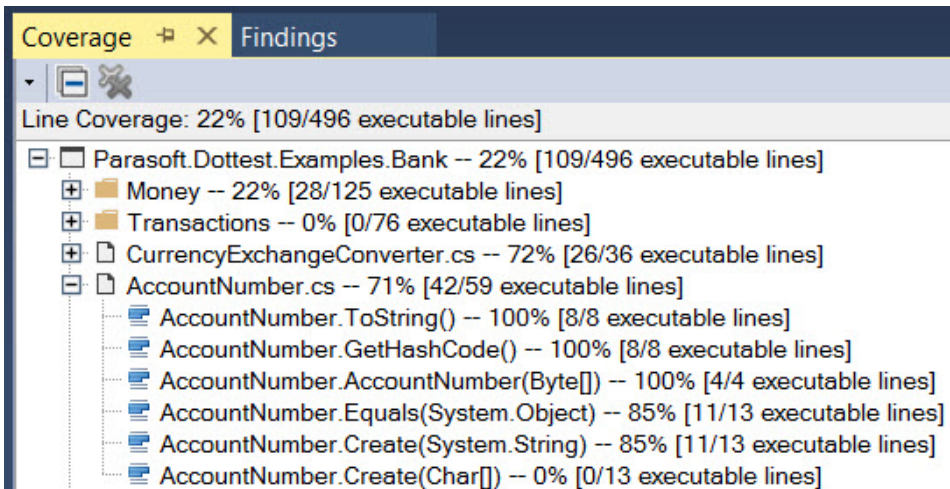
Sort by: You can sort by file name or percentage of coverage in ascending or descending order.

Merge coverage: Enabling this option combines coverage information when additional coverage data is imported; disabling this option clears the coverage view and replaces it with the new data.

4. Import coverage data from an XML file or from DTP (see [Importing Coverage](#)). The Coverage view will show the percentage of coverage, as well as how many lines of code are covered with tests of the available lines in the file.

Jtest automatically matches imported coverage data with resources available in your workspace. As a result, the structure of resources displayed in the Coverage view is identical with the project structure displayed in the IDE.

If you import coverage associated with resources that were modified or are no longer available in your workspace, the coverage data is recalculated to reflect the current coverage status for resources in your workspace.



5. Click a file to view the classes in the file.
6. Click a class to open it in the editor and display the methods.
7. Click a method to view it in the editor with color-coded markers that indicate coverage.

```
public Currency Convert(Currency input, CurrencyInfo target)
{
    if (input.CurrencyInfo == target)
    {
        return input;
    }
    else
    {
        Currency main = ConvertToMainCurrency(input);
        return ConvertFromMainCurrency(main, target);
    }
}

private Currency ConvertToMainCurrency(Currency input)
{
    if (input.CurrencyInfo == _mainCurrency)
    {
        return input;
    }
    else
    {
        decimal value = input.GetDecimalValue() / (decimal)_ratios[input.CurrencyInfo];
        return CurrencyProvider.GetCurrency(value, _mainCurrency);
    }
}

private Currency ConvertFromMainCurrency(Currency input, CurrencyInfo target)
{
    if (input == null)
    {

```

Coverage

Line Coverage: 22% [109/496 executable lines]

- Parasoft.DotTest.Examples.Bank -- 22% [109/496 executable lines]
 - Money -- 22% [28/125 executable lines]
 - Transactions -- 0% [0/76 executable lines]
 - CurrencyExchangeConverter.cs -- 72% [26/36 executable lines]
 - CurrencyExchangeConverter.CurrencyExchangeConverter(Parasoft.DotTest.Examples.Bank.Money.CurrencyInfo) -- 100% [5/5 executable lines]
 - CurrencyExchangeConverter.AddRatio(Parasoft.DotTest.Examples.Bank.Money.CurrencyInfo, System.Single) -- 100% [3/3 executable lines]
 - CurrencyExchangeConverter.ConvertToMainCurrency(Parasoft.DotTest.Examples.Bank.Money.Currency) -- 75% [6/8 executable lines]
 - CurrencyExchangeConverter.Convert(Parasoft.DotTest.Examples.Bank.Money.Currency, Parasoft.DotTest.Examples.Bank.Money.CurrencyInfo) -- 75% [6/8

- Covered lines are flagged with a green marker.
- Code that is not covered is flagged with a red marker.