



# Core Testing>Basic Testing>Day 9

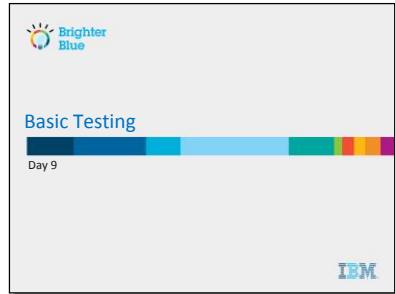
Student Guide

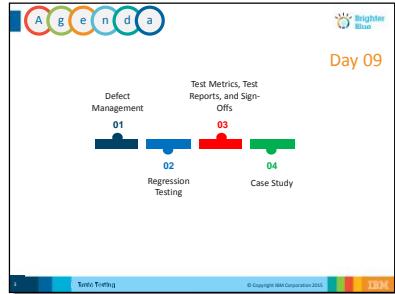


## Contents

<u>MODULE 01: DEFECT MANAGEMENT</u>	<u>3</u>
<u>MODULE 02: REGRESSION TESTING</u>	<u>41</u>
<u>MODULE 03: TEST METRICS, TEST REPORTS, AND SIGN-OFF</u>	<u>50</u>
<u>MODULE 04: CASE STUDY</u>	<u>83</u>

## Module 01: Defect Management

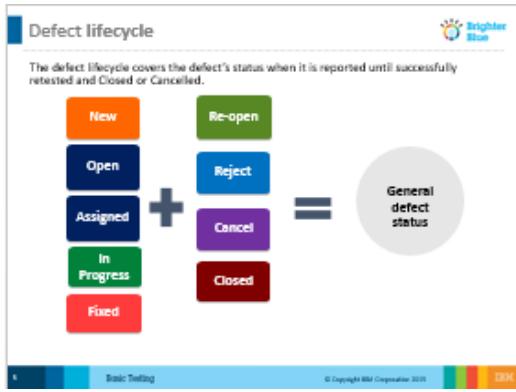
Slide Content	Use this space for your own notes
Slide 1 	

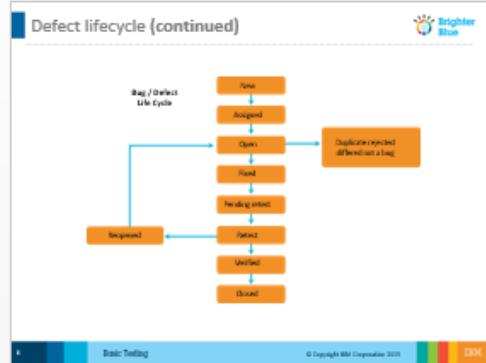
Slide Content	Use this space for your own notes
<p>Slide 2</p>  <p>At the end of the module, you should be able to:</p> <ul style="list-style-type: none"> <li>▪ Define defect</li> <li>▪ Define and illustrate defect life cycle</li> <li>▪ Describe the testing principles</li> <li>▪ Define and classify defect classes</li> <li>▪ List the requirement specification defects</li> <li>▪ Describe the different types of design defects</li> <li>▪ List the design and coding defects</li> <li>▪ Illustrate defect origination and the cost of fixing defects</li> <li>▪ Describe how to prevent defects, the cost of errors and the legal consequences of defective testing</li> <li>▪ Illustrate defect and change tracking</li> <li>▪ Describe how to conduct log change requests</li> </ul>	

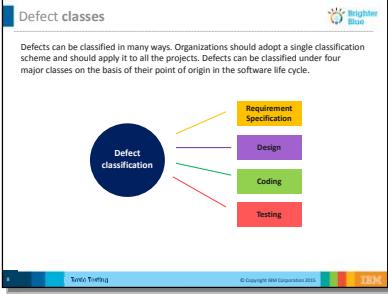
Slide Content	Use this space for your own notes
<ul style="list-style-type: none"><li>▪ Describe what are test defect metrics, defect severity, defect find and fix rate, and other defect metrics</li><li>▪ Describe the common defect tracking tools</li><li>▪ Illustrate the common defect or change request life cycle</li><li>▪ Describe what are defect remarks and how to use checklist before entering a defect and how to avoid duplication</li></ul>	

# Core Testing > Basic Testing > Day 9

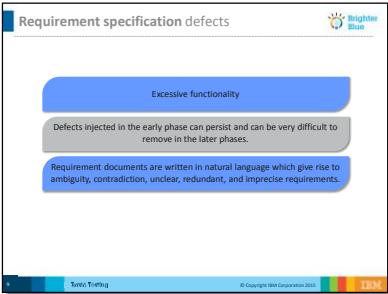
Slide Content	Use this space for your own notes
<p>Slide 3</p> <div data-bbox="460 409 958 796"><p>What is a defect?</p><p>A Software Defect / Bug is a condition in a software product which does not meet a software requirement (as stated in the requirement specification) or end-user expectations (which may not be specified but are reasonable). In other words, a defect is an error in coding or logic that breaks and program fails to perform or malfunction or produces incorrect/unexpected results.</p><p>Basic Testing</p><p>© Copyright IBM Corporation 2008</p><p>IBM</p></div>	

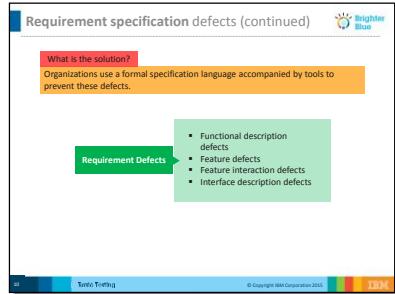
Slide Content	Use this space for your own notes
<p>Slide 4</p> <div data-bbox="451 404 967 796"><p>Defect lifecycle</p><p>The defect lifecycle covers the defect's status when it is reported until successfully retested and Closed or Cancelled.</p><p>A diagram titled "Defect lifecycle" showing the states a defect can go through. On the left, a vertical stack of boxes represents initial states: New (orange), Open (dark blue), Assigned (medium blue), In Progress (green), and Fixed (red). An arrow points from this stack to a central column of boxes: Re-open (light green), Reject (light blue), Cancel (purple), and Closed (dark red). A plus sign is between the two columns, and an equals sign is to the right of the central column, pointing to a large grey circle labeled "General defect status".</p><p>Basic Testing</p><p>© Copyright IBM Corporation 2010</p><p>IBM</p></div>	

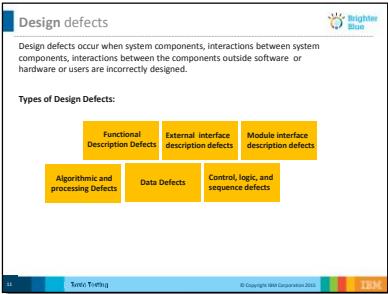
Slide Content	Use this space for your own notes
<p>Slide 5</p>  <p>The diagram illustrates the Defect lifecycle (continued). It shows a sequence of states: New, Assigned, Open, Fixed, Pending review, Tested, Verified, and Closed. A feedback loop labeled "Impact" connects the "Closed" state back to the "New" state. Additionally, an arrow points from the "Open" state to a box labeled "Defects reported off defect in a log".</p>	

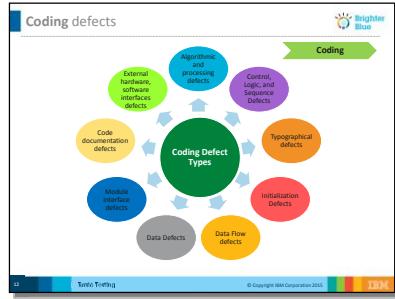
Slide Content	Use this space for your own notes
<p>Slide 6</p>  <p>The diagram illustrates the classification of defects. A central dark blue circle labeled "Defect classification" has four arrows pointing to colored boxes: "Requirement Specification" (yellow), "Design" (purple), "Coding" (green), and "Testing" (red).</p>	

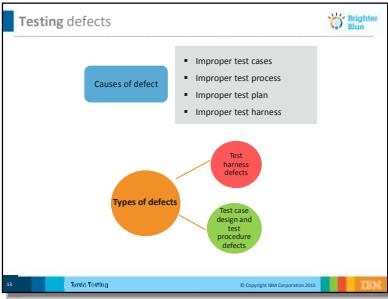
Slide Content	Use this space for your own notes
<p>Slide 7</p> <div data-bbox="451 404 967 791"><p>Defect: root cause analysis</p><p>Root cause analysis (RCA) is a method of problems solving used for identifying the root causes of faults or problems.</p><p>How it helps</p><ul style="list-style-type: none"><li>Help identify the root cause of a problem.</li><li>Determine the relationship between different root causes of a problem.</li><li>One of the simplest tools; easy to complete without statistical analysis.</li></ul><p>Basic Testing</p><p>© Copyright IBM Corporation 2008</p></div>	

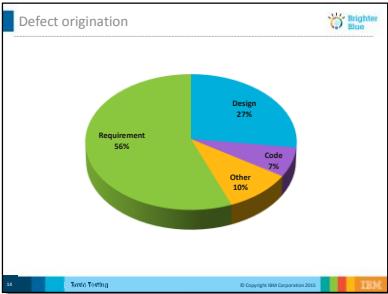
Slide Content	Use this space for your own notes
<p>Slide 8</p>  <p>The slide content is titled "Requirement specification defects" and includes the following points:</p> <ul style="list-style-type: none"><li>Excessive functionality</li><li>Defects injected in the early phase can persist and can be very difficult to remove in the later phases.</li><li>Requirement documents are written in natural language which give rise to ambiguity, contradiction, unclear, redundant, and imprecise requirements.</li></ul> <p>At the bottom of the slide, there is footer text: "© Copyright IBM Corporation 2012" and the "IBM" logo.</p>	

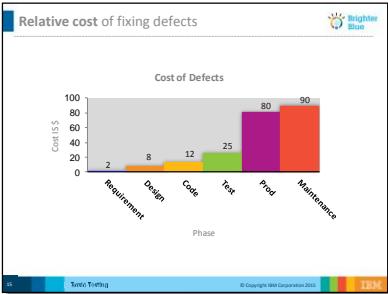
Slide Content	Use this space for your own notes
<p>Slide 9</p>  <p>The slide is titled "Requirement specification defects (continued)". It features a red box at the top left containing the question "What is the solution?". Below it, a yellow box states: "Organizations use a formal specification language accompanied by tools to prevent these defects." A green callout box labeled "Requirement Defects" contains a bulleted list: "Functional description defects", "Feature defects", "Feature interaction defects", and "Interface description defects". The bottom of the slide includes the "Testing" navigation bar, the copyright notice "© Copyright IBM Corporation 2012", and the IBM logo.</p>	

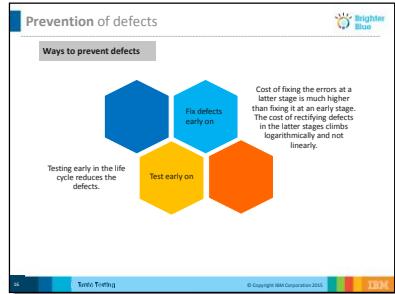
Slide Content	Use this space for your own notes
<p>Slide 10</p>  <p>The slide is titled "Design defects". It defines design defects as occurring when system components, interactions between system components, interactions between the components outside software or hardware or users are incorrectly designed. It lists several types of design defects:</p> <ul style="list-style-type: none"><li>Functional Description Defects</li><li>External interface description defects</li><li>Module interface description defects</li><li>Algorithmic and processing Defects</li><li>Data Defects</li><li>Control, logic, and sequence defects</li></ul> <p>At the bottom, there is a navigation bar with icons for back, forward, search, and other presentation controls.</p>	

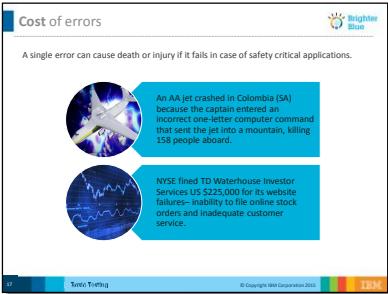
Slide Content	Use this space for your own notes
<p>Slide 11</p>  <p>The diagram is titled "Coding defects" and features a central circle labeled "Coding Defect Types". Surrounding this center are ten smaller circles, each representing a specific type of defect, connected by arrows pointing towards the central circle. The types listed are: External hardware/software anomalies defects (green), Algorithm and processing defects (light blue), Control, timing and Sequence Defects (purple), Topographical defects (orange), Initialization Defects (red), Data Flow defects (yellow), Data Defects (grey), Module interface defects (blue), Code documentation defects (orange), and External hardware/software anomalies defects (green). A green arrow points from the top right towards the central circle. The bottom of the slide includes a footer bar with the text "Testing Testing", "© Copyright IBM Corporation 2012", and the IBM logo.</p> <p>Coding defects are derived from errors in implementing the code. Coding defects are similar to design defects.</p>	

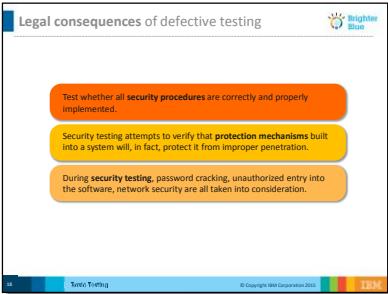
Slide Content	Use this space for your own notes
<p>Slide 12</p>  <p>The diagram illustrates the causes and types of defects in testing. It shows a central orange circle labeled 'Types of defects' connected by arrows to two smaller circles: a red one labeled 'Test harness defects' and a green one labeled 'Test case design and test procedure defects'. Above this, a blue box labeled 'Causes of defect' contains a bulleted list: 'Improper test cases', 'Improper test process', 'Improper test plan', and 'Improper test harness'.</p> <p>Defects originate due to improper Test Plan, Test Cases, Test Harness, and Test Process.</p>	

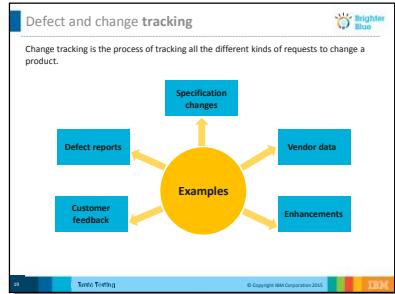
Slide Content	Use this space for your own notes										
<p>Slide 13</p>  <p>The pie chart illustrates the distribution of defects by their origin:</p> <table border="1"><thead><tr><th>Category</th><th>Percentage</th></tr></thead><tbody><tr><td>Requirement</td><td>56%</td></tr><tr><td>Design</td><td>27%</td></tr><tr><td>Code</td><td>7%</td></tr><tr><td>Other</td><td>10%</td></tr></tbody></table> <p>© Copyright IBM Corporation 2012</p>	Category	Percentage	Requirement	56%	Design	27%	Code	7%	Other	10%	
Category	Percentage										
Requirement	56%										
Design	27%										
Code	7%										
Other	10%										

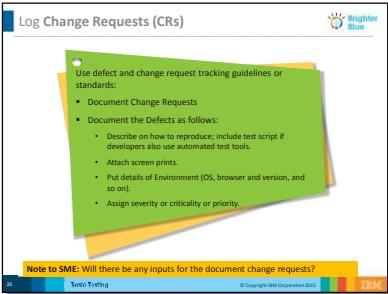
Slide Content	Use this space for your own notes														
<p>Slide 14</p>  <p>The chart illustrates the cost of fixing defects at various stages of the software development process. The Y-axis represents the cost in dollars, ranging from 0 to 100. The X-axis lists the phases: Requirement, Design, Code, Test, Prod, and Maintenance. The cost increases significantly as the defect reaches the production and maintenance stages.</p> <table border="1"><thead><tr><th>Phase</th><th>Cost (\$)</th></tr></thead><tbody><tr><td>Requirement</td><td>2</td></tr><tr><td>Design</td><td>8</td></tr><tr><td>Code</td><td>12</td></tr><tr><td>Test</td><td>25</td></tr><tr><td>Prod</td><td>80</td></tr><tr><td>Maintenance</td><td>90</td></tr></tbody></table>	Phase	Cost (\$)	Requirement	2	Design	8	Code	12	Test	25	Prod	80	Maintenance	90	
Phase	Cost (\$)														
Requirement	2														
Design	8														
Code	12														
Test	25														
Prod	80														
Maintenance	90														

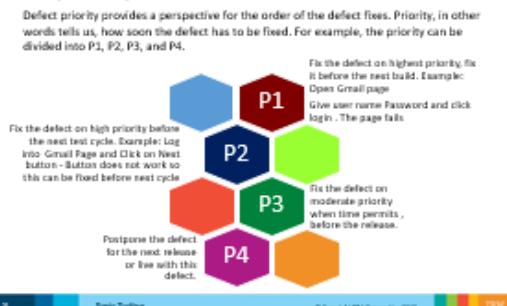
Slide Content	Use this space for your own notes
<p>Slide 15</p>  <p>The slide is titled "Prevention of defects" and includes the "Ways to prevent defects" section. It features three hexagonal icons: a blue one labeled "Fix defects early on", a yellow one labeled "Test early on", and an orange one labeled "Testing early in the life cycle reduces the defects.". To the right of the icons, there is explanatory text: "Cost of fixing the errors at a later stage is much higher than fixing it at an early stage. The cost of rectifying defects in the software climbs logarithmically and not linearly." The slide footer includes the "© Copyright IBM Corporation 2012" and "IBM" logos.</p>	

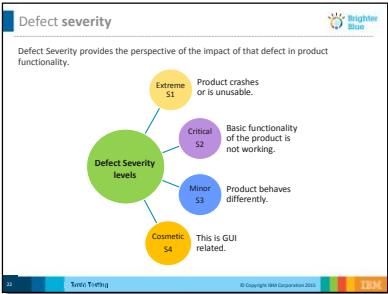
Slide Content	Use this space for your own notes
<p>Slide 16</p>  <p>The slide is titled "Cost of errors" and includes the Brighter Blue logo. It states: "A single error can cause death or injury if it fails in case of safety critical applications." Two examples are shown: 1) An AA jet crashed in Colombia (SA) because the captain entered an incorrect one-letter computer command that sent the jet into a mountain, killing 158 people aboard. 2) NYSE fined TD Waterhouse Investor Services US \$225,000 for its website failures— inability to file online stock orders and inadequate customer service.</p>	

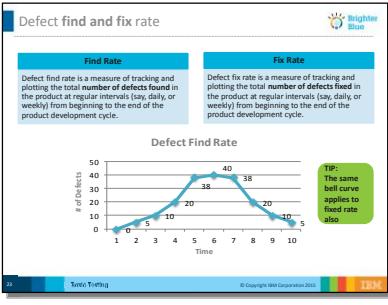
Slide Content	Use this space for your own notes
<p>Slide 17</p> 	

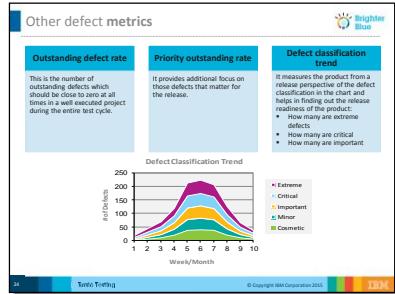
Slide Content	Use this space for your own notes
<p>Slide 18</p>  <p>The diagram illustrates the concept of Defect and change tracking. It features a central yellow circle labeled "Examples" with five arrows pointing to surrounding blue boxes. The boxes are labeled: "Defect reports", "Specification changes", "Vendor data", "Enhancements", and "Customer feedback". The entire slide has a dark blue header bar at the top.</p>	

Slide Content	Use this space for your own notes
<p>Slide 19</p>  <p>The slide content is as follows:</p> <p><b>Log Change Requests (CRs)</b></p> <p>Use defect and change request tracking guidelines or standards:</p> <ul style="list-style-type: none"><li>• Document Change Requests</li><li>• Document the Defects as follows:<ul style="list-style-type: none"><li>• Describe on how to reproduce; include test script if developers also use automated test tools.</li><li>• Attach screen prints.</li><li>• Put details of Environment (OS, browser and version, and so on).</li><li>• Assign severity or criticality or priority.</li></ul></li></ul> <p>Note to SME: Will there be any inputs for the document change requests?</p> <p>20 Teste Testing © Copyright IBM Corporation 2012 IBM</p>	

Slide Content	Use this space for your own notes
<p>Slide 20</p> <div style="border: 1px solid #ccc; padding: 10px;"> <p><b>Test defect metrics</b></p> <p><b>Priority and severity</b></p> <p>Defect priority provides a perspective for the order of the defect fixes. Priority, in other words tells us, how soon the defect has to be fixed. For example, the priority can be divided into P1, P2, P3, and P4.</p>  <ul style="list-style-type: none"> <li>P1: Fix the defect on highest priority, fix it before the next build. Example: Open Gmail page. Give user name Password and click login. The page fails.</li> <li>P2: Fix the defect on high priority before the next test cycle. Example: Log into Gmail Page and Click on Next button - Button does not work so this can be fixed before next cycle.</li> <li>P3: Fix the defect on moderate priority when time permits, before the release.</li> <li>P4: Postpone the defect for the next release or live with this defect.</li> </ul> <p>Basic Testing</p> <p>© Copyright IBM Corporation 2010</p> </div>	

Slide Content	Use this space for your own notes
<p>Slide 21</p>  <p>The diagram illustrates the four levels of defect severity:</p> <ul style="list-style-type: none"><li><b>Extreme S1</b>: Product crashes or is unusable.</li><li><b>Critical S2</b>: Basic functionality of the product is not working.</li><li><b>Minor S3</b>: Product behaves differently.</li><li><b>Cosmetic S4</b>: This is GUI related.</li></ul>	

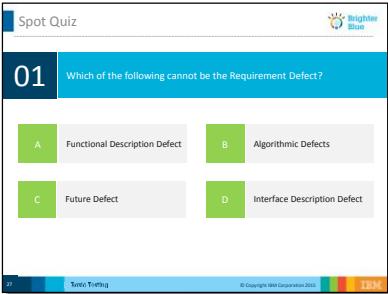
Slide Content	Use this space for your own notes																						
<p>Slide 22</p>  <p>The slide contains two definitions: 'Find Rate' and 'Fix Rate'. Below them is a line graph titled 'Defect Find Rate' showing the number of defects found over time. A callout box points to the peak of the bell curve on the graph with the text: 'TOP: The same bell curve applies to find rate data'.</p> <table border="1"> <thead> <tr> <th>Time</th> <th>Number of Defects</th> </tr> </thead> <tbody> <tr><td>1</td><td>0</td></tr> <tr><td>2</td><td>0</td></tr> <tr><td>3</td><td>5</td></tr> <tr><td>4</td><td>10</td></tr> <tr><td>5</td><td>20</td></tr> <tr><td>6</td><td>40</td></tr> <tr><td>7</td><td>38</td></tr> <tr><td>8</td><td>20</td></tr> <tr><td>9</td><td>10</td></tr> <tr><td>10</td><td>5</td></tr> </tbody> </table>	Time	Number of Defects	1	0	2	0	3	5	4	10	5	20	6	40	7	38	8	20	9	10	10	5	
Time	Number of Defects																						
1	0																						
2	0																						
3	5																						
4	10																						
5	20																						
6	40																						
7	38																						
8	20																						
9	10																						
10	5																						

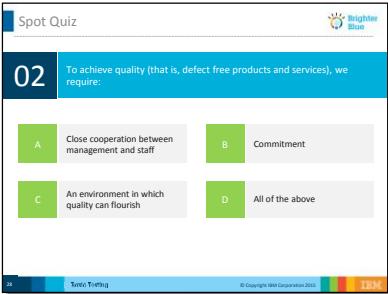
Slide Content	Use this space for your own notes
<p>Slide 23</p>  <p><b>Outstanding defect rate</b> This is the number of outstanding defects which should be zero at all times in a well executed project during the entire test cycle.</p> <p><b>Priority outstanding rate</b> It provides additional focus on those defects that matter for the release.</p> <p><b>Defect classification trend</b> It measures the product from a perspective of defect classification in the chart and helps in finding out the release risk of the product.</p> <ul style="list-style-type: none"> <li>• How many are extreme</li> <li>• How many are important</li> <li>• How many are critical</li> <li>• How many are important</li> </ul> <p>Defect Classification Trend</p> <p>Defects</p> <p>Weeks/Month</p> <p>Legend: Extreme (dark purple), Critical (purple), Important (blue), Minor (light blue), Cosmetic (yellow)</p> <p>26 Testin Testing © Copyright IBM Corporation 2012 IBM</p>	

# Core Testing > Basic Testing > Day 9

Slide Content	Use this space for your own notes
<p>Slide 24</p>  <p>The slide content displays a grid of six common defect tracking tools:</p> <ul style="list-style-type: none"><li>BugZilla: Available for free download at <a href="http://www.bugzilla.com">www.bugzilla.com</a></li><li>Test Track Pro: From Sea Pine</li><li>IBM Rational clearquest</li><li>Track Gear: from Logi Gear Corporation</li><li>PR Tracker: <a href="http://www.prtracker.com">www.prtracker.com</a></li><li>HR-Quality Center: Defect module (Licensed tool)</li></ul> <p>Navigation icons at the bottom include arrows for previous/next slides, a search icon, and the IBM logo.</p>	

Slide Content	Use this space for your own notes
<p>Slide 25</p> 	

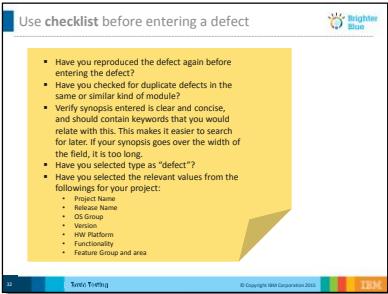
Slide Content	Use this space for your own notes
<p>Slide 26</p>  <p>The slide displays a question from a 'Spot Quiz' titled '01'. The question asks: 'Which of the following cannot be the Requirement Defect?'. The options are: A) Functional Description Defect, B) Algorithmic Defects, C) Future Defect, and D) Interface Description Defect. The slide includes a navigation bar at the bottom with icons for back, forward, search, and help, along with the IBM logo.</p>	

Slide Content	Use this space for your own notes
<p>Slide 27</p>  <p>The slide displays a 'Spot Quiz' interface. At the top left is a blue square icon with a white '02'. To its right is the text 'To achieve quality (that is, defect free products and services), we require:' followed by a question mark. Below this, there are four options labeled A, B, C, and D, each in a green box:</p> <ul style="list-style-type: none"><li>A Close cooperation between management and staff</li><li>B Commitment</li><li>C An environment in which quality can flourish</li><li>D All of the above</li></ul> <p>At the bottom of the slide, there is a navigation bar with icons for back, forward, and search, along with the text '© Copyright IBM Corporation 2012' and the IBM logo.</p>	

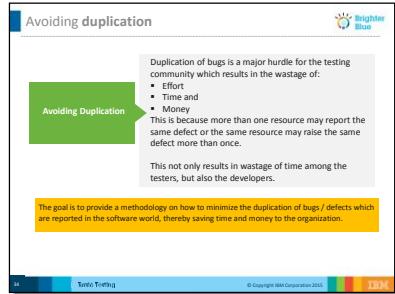
Slide Content	Use this space for your own notes
<p>Slide 28</p>  <p>The slide displays a question from a 'Spot Quiz' titled '03'. The question is: 'Defect is any variance between actual and _____ results.' Below the question are four options:</p> <ul style="list-style-type: none"><li>A Expected</li><li>B True</li><li>C Perfect</li><li>D Wrong</li></ul> <p>At the bottom of the slide, there is footer text: '© Copyright IBM Corporation 2012' and the 'IBM' logo.</p>	

Slide Content	Use this space for your own notes
<p>Slide 29</p>  <ul style="list-style-type: none"> <li>▪ <b>Condense:</b> Say it clearly but briefly.</li> <li>▪ <b>Accurate:</b> Is it a defect or could it be user error, misunderstanding, and so on?</li> <li>▪ <b>Neutralize:</b> Just the facts. No zingers. No humor. No emotion.</li> <li>▪ <b>Precise:</b> Explicitly, what is the problem?</li> <li>▪ <b>Isolate:</b> What has been done to isolate the problem?</li> <li>▪ <b>Re-create:</b> What are the essentials in triggering/re-creating this problem? (environment, steps, conditions)</li> <li>▪ <b>Impact:</b> What is the impact to the customer? What is the impact to testing? Sell the defect.</li> <li>▪ <b>Debug:</b> What does development need to make it easier to debug? (Traces, dumps, logs, immediate access, and so on.)</li> <li>▪ <b>Evidence:</b> What documentation will prove the existence of the error?</li> </ul>	

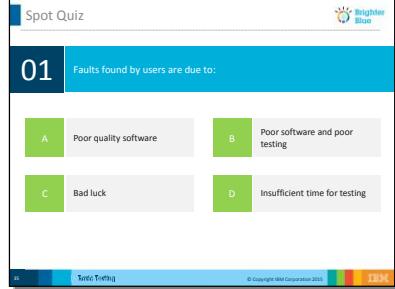
Slide Content	Use this space for your own notes

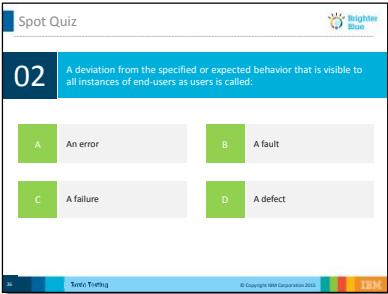
Slide Content	Use this space for your own notes
<p>Slide 30</p>  <p>The screenshot shows a slide titled "Use checklist before entering a defect". It contains a yellow sticky-note-like callout with the following checklist:</p> <ul style="list-style-type: none"><li>Have you reproduced the defect again before entering the defect?</li><li>Have you checked for duplicate defects in the same or similar kind of module?</li><li>Verify synopsis entered is clear and concise, and should contain keywords that you would relate with this. This makes it easier to search for later if your synopsis goes over the width of the field, if too long.</li><li>Have you selected type as "defect"?</li><li>Have you selected the relevant values from the followings for your project:<ul style="list-style-type: none"><li>Project Name</li><li>Module Name</li><li>OS Group</li><li>Category</li><li>HW Platform</li><li>Functionality</li><li>Feature Group and area</li></ul></li></ul> <p>At the bottom of the slide, there are navigation buttons for "Testin Testing" and "IBM".</p>	

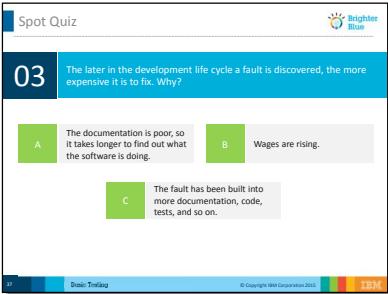
Slide Content	Use this space for your own notes
<p>Slide 31</p> <div data-bbox="445 404 967 796"><p>Use checklist before entering a defect (continued) </p><ul style="list-style-type: none"><li>▪ Have you selected the most appropriate severity of the defect based on the defect complexity?</li><li>▪ Have you entered the description for:<ul style="list-style-type: none"><li>▪ Expected behavior</li><li>▪ Observed behavior</li></ul></li><li>▪ Have you entered the description for steps to reproduce to contain detailed reproductive steps so that reviewers or developers will not request for more information?</li><li>▪ What are the builds used for testing?</li><li>▪ What about connection information like database, server name, and so on?</li><li>▪ Have you attached screenshot(s), log file(s), trace file(s), sample report(s), Sample application(s) for defect?</li><li>▪ Have you verified that there are no typo error within defect entry?</li></ul><p>  © Copyright IBM Corporation 2010 </p></div>	

Slide Content	Use this space for your own notes
<p>Slide 32</p>  <p>The slide content is as follows:</p> <p><b>Avoiding duplication</b></p> <p>Duplication of bugs is a major hurdle for the testing community which results in the wastage of:</p> <ul style="list-style-type: none"><li>Effort</li><li>Time and</li><li>Money</li></ul> <p>This is because more than one resource may report the same defect or the same resource may raise the same defect more than once.</p> <p>This not only results in wastage of time among the testers, but also the developers.</p> <p>The goal is to provide a methodology on how to minimize the duplication of bugs / defects which are reported in the software world, thereby saving time and money to the organization.</p> <p>Source: Testing Simplified © Copyright IBM Corporation 2012 IBM</p>	

Slide Content	Use this space for your own notes
Slide 33 	
Slide 34	

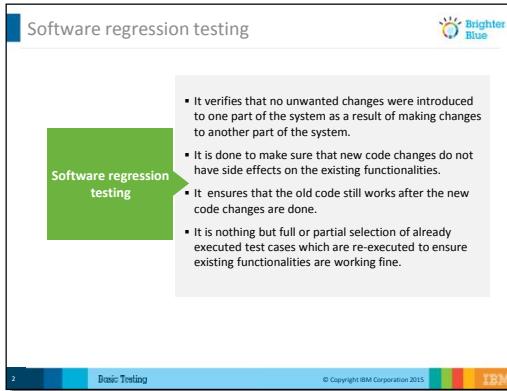
Slide Content	Use this space for your own notes
 <p>01 Faults found by users are due to:</p> <p>A Poor quality software      B Poor software and poor testing</p> <p>C Bad luck      D Insufficient time for testing</p>	

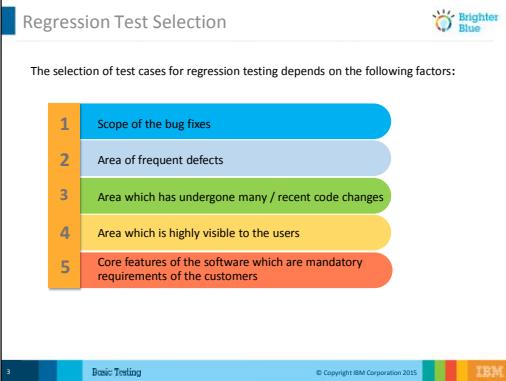
Slide Content	Use this space for your own notes
<p>Slide 35</p>  <p>The slide displays a question from a 'Spot Quiz' titled '02'. The question asks: 'A deviation from the specified or expected behavior that is visible to all instances of end-users as users is called:'. Below the question are four options: A) An error, B) A fault, C) A failure, and D) A defect. The correct answer is not explicitly shown.</p>	

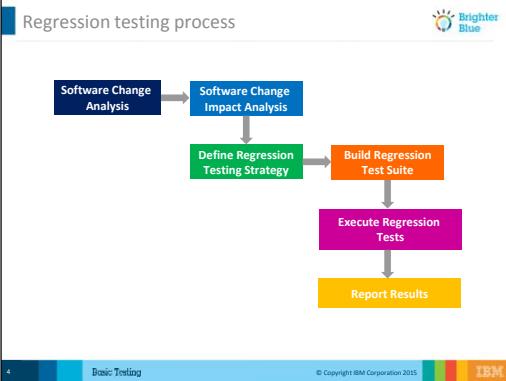
Slide Content	Use this space for your own notes
<p>Slide 36</p>  <p>The later in the development life cycle a fault is discovered, the more expensive it is to fix. Why?</p> <p>A The documentation is poor, so it takes longer to find out what the software is doing.</p> <p>B Wages are rising.</p> <p>C The fault has been built into more documentation, code, tests, and so on.</p>	

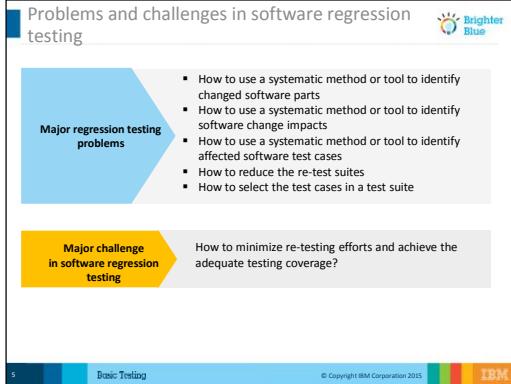
## Module 02: Regression Testing

Slide Content	Use this space for your own notes
<p>Slide 37</p>  <p>The objectives of this module are to:</p> <ul style="list-style-type: none"><li>▪ Define and describe the importance and pre-requisites of Test Case writing</li><li>▪ Describe the characteristics of a good Test Case and how to write it</li><li>▪ List the attributes of Test Case and the documents required to write a Test case</li></ul>	

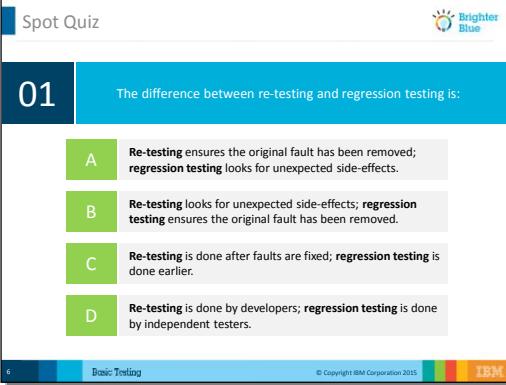
Slide Content	Use this space for your own notes
Slide 38  <p>The slide content is titled "Software regression testing". It includes a green sidebar with the title and a main text area with a bulleted list:</p> <ul style="list-style-type: none"><li>▪ It verifies that no unwanted changes were introduced to one part of the system as a result of making changes to another part of the system.</li><li>▪ It is done to make sure that new code changes do not have side effects on the existing functionalities.</li><li>▪ It ensures that the old code still works after the new code changes are done.</li><li>▪ It is nothing but full or partial selection of already executed test cases which are re-executed to ensure existing functionalities are working fine.</li></ul> <p>At the bottom, there is a navigation bar with icons for back, forward, and search, followed by the text "Basic Testing" and the IBM logo.</p>	
Slide 39	

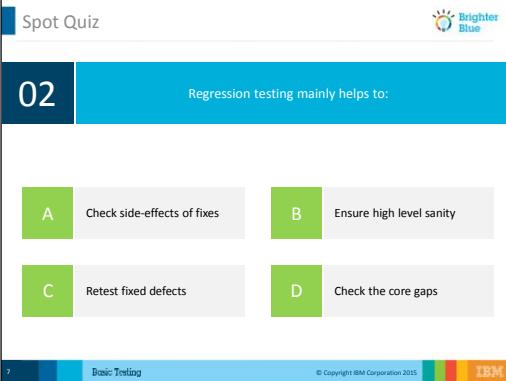
Slide Content	Use this space for your own notes
 <p>The selection of test cases for regression testing depends on the following factors:</p> <ol style="list-style-type: none"><li>1 Scope of the bug fixes</li><li>2 Area of frequent defects</li><li>3 Area which has undergone many / recent code changes</li><li>4 Area which is highly visible to the users</li><li>5 Core features of the software which are mandatory requirements of the customers</li></ol> <p>Basic Testing © Copyright IBM Corporation 2015 IBM</p>	
Slide 40	

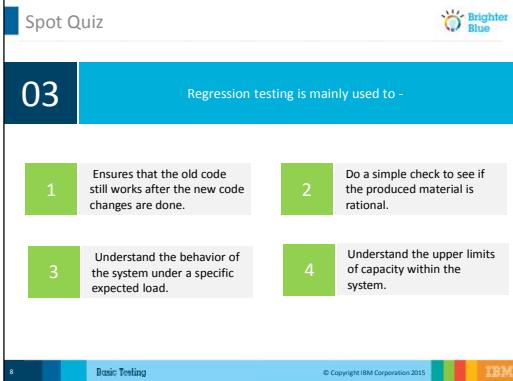
Slide Content	Use this space for your own notes
 <p>The diagram illustrates the Regression testing process flow:</p> <pre>graph TD; A[Software Change Analysis] --&gt; B[Software Change Impact Analysis]; B --&gt; C[Define Regression Testing Strategy]; C --&gt; D[Build Regression Test Suite]; D --&gt; E[Execute Regression Tests]; E --&gt; F[Report Results]</pre> <p>Annotations on the slide:</p> <ul style="list-style-type: none"><li>Top right corner: Brighter Blue logo</li><li>Bottom left: Basic Testing</li><li>Bottom center: © Copyright IBM Corporation 2015</li><li>Bottom right: IBM logo</li></ul>	
Slide 41	

Slide Content	Use this space for your own notes
 <p>The slide content is titled "Problems and challenges in software regression testing". It features a blue arrow pointing right labeled "Major regression testing problems" containing a bulleted list of six items. To the right of the arrow is a yellow arrow pointing right labeled "Major challenge in software regression testing" with the text "How to minimize re-testing efforts and achieve the adequate testing coverage?". The slide footer includes the number "5", the title "Basic Testing", and the IBM logo.</p>	
Slide 42	

Slide Content	Use this space for your own notes
 <p>Slide 43</p>	

Slide Content	Use this space for your own notes
 <p>The difference between re-testing and regression testing is:</p> <ul style="list-style-type: none"><li>A Re-testing ensures the original fault has been removed; regression testing looks for unexpected side-effects.</li><li>B Re-testing looks for unexpected side-effects; regression testing ensures the original fault has been removed.</li><li>C Re-testing is done after faults are fixed; regression testing is done earlier.</li><li>D Re-testing is done by developers; regression testing is done by independent testers.</li></ul>	
Slide 44	

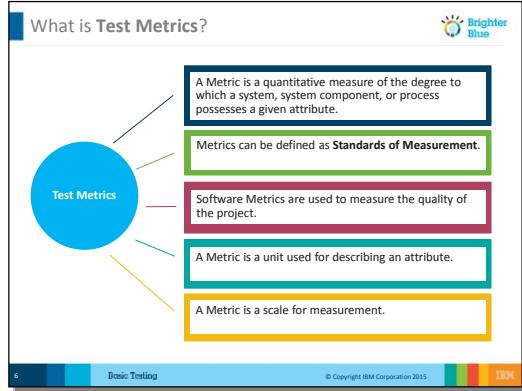
Slide Content	Use this space for your own notes
 <p>Spot Quiz</p> <p>02 Regression testing mainly helps to:</p> <p>A Check side-effects of fixes      B Ensure high level sanity</p> <p>C Retest fixed defects      D Check the core gaps</p> <p>7 Basic Testing © Copyright IBM Corporation 2015 IBM</p>	
Slide 45	

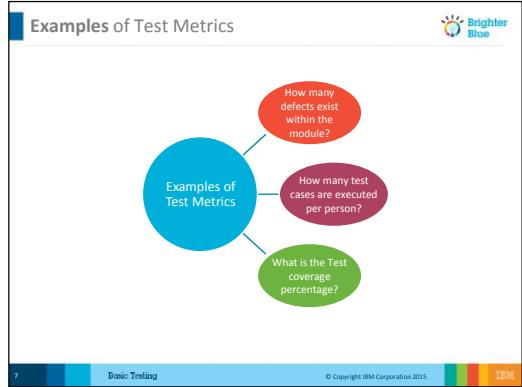
Slide Content	Use this space for your own notes
 <p>Spot Quiz</p> <p>03 Regression testing is mainly used to -</p> <ul style="list-style-type: none"><li>1 Ensures that the old code still works after the new code changes are done.</li><li>2 Do a simple check to see if the produced material is rational.</li><li>3 Understand the behavior of the system under a specific expected load.</li><li>4 Understand the upper limits of capacity within the system.</li></ul> <p>Basic Testing © Copyright IBM Corporation 2015 IBM</p>	

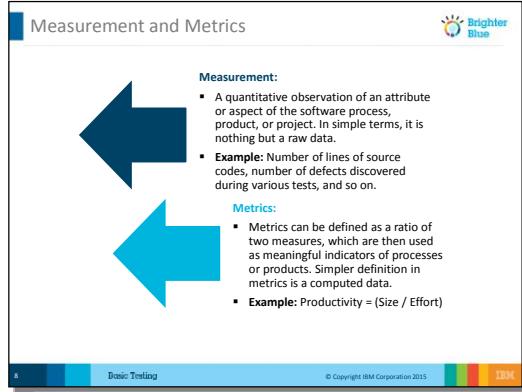
## Module 03: Test Metrics, Test Reports, and Sign-off

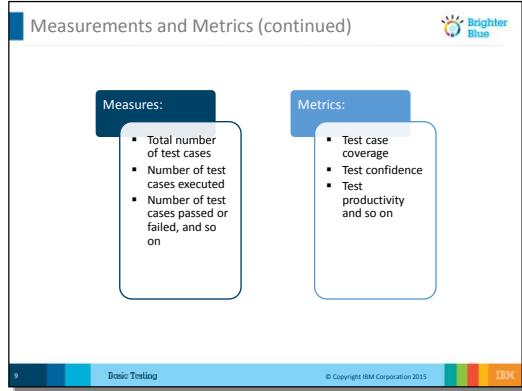
Slide Content	Use this space for your own notes
<p>Slide 46</p>  <p>The slide thumbnail shows a white background with a dark blue header bar. The header bar contains the number '03' on the left, followed by the text 'Test Metrics, Test Reports, and Sign-off' in white. To the right of the text is a decorative bar composed of vertical colored stripes in blue, green, yellow, and red. At the bottom right of the slide is the word 'IBM'.</p> <p>At the end of this module, you should be able to:</p> <ul style="list-style-type: none"><li>▪ Define Test Metrics</li><li>▪ Recall the difference between Measurements and Metrics</li><li>▪ Identify the significance of Software Testing Metrics</li><li>▪ List the benefits of Metrics</li><li>▪ Explain The Metrics Life Cycle</li><li>▪ Recognize the types of Test Reports, such as Test Execution And Test Summary Report</li><li>▪ Describe Sign-offs</li></ul>	

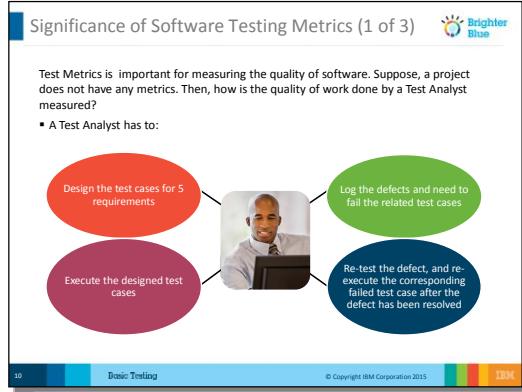
Slide Content	Use this space for your own notes

Slide Content	Use this space for your own notes
<p>Slide 47</p>  <p>The slide is titled "What is Test Metrics?" and features a central blue circle labeled "Test Metrics". Five lines radiate from this circle to five rectangular boxes, each containing a definition of a metric:</p> <ul style="list-style-type: none"><li>A Metric is a quantitative measure of the degree to which a system, system component, or process possesses a given attribute.</li><li>Metrics can be defined as Standards of Measurement.</li><li>Software Metrics are used to measure the quality of the project.</li><li>A Metric is a unit used for describing an attribute.</li><li>A Metric is a scale for measurement.</li></ul> <p>At the bottom of the slide, there is a navigation bar with icons for back, forward, and search, followed by the text "Basic Testing" and "© Copyright IBM Corporation 2015 IBM".</p>	

Slide Content	Use this space for your own notes
<p>Slide 48</p>  <p>The diagram illustrates 'Examples of Test Metrics' with three colored circles (red, purple, and green) connected to a central blue circle labeled 'Examples of Test Metrics'. The red circle contains the text 'How many defects exist within the module?'. The purple circle contains the text 'How many test cases are executed per person?'. The green circle contains the text 'What is the Test coverage percentage?'. The background of the slide features a grid pattern.</p>	

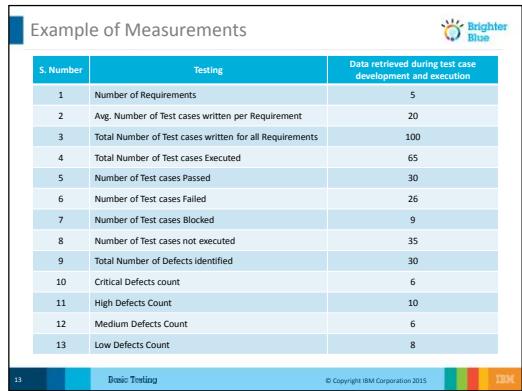
Slide Content	Use this space for your own notes
<p>Slide 49</p>  <p><b>Measurement:</b></p> <ul style="list-style-type: none"><li>▪ A quantitative observation of an attribute or aspect of the software process, product, or project. In simple terms, it is nothing but a raw data.</li><li>▪ <b>Example:</b> Number of lines of source codes, number of defects discovered during various tests, and so on.</li></ul> <p><b>Metrics:</b></p> <ul style="list-style-type: none"><li>▪ Metrics can be defined as a ratio of two measures, which are then used as meaningful indicators of processes or products. Simpler definition in metrics is a computed data.</li><li>▪ <b>Example:</b> Productivity = (Size / Effort)</li></ul> <p>Basic Testing © Copyright IBM Corporation 2015 IBM</p>	

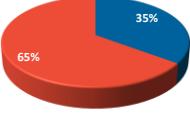
Slide Content	Use this space for your own notes
<p>Slide 50</p> 	

Slide Content	Use this space for your own notes
<p>Slide 51</p>  <p>The slide is titled "Significance of Software Testing Metrics (1 of 3)" and features a flowchart with a central figure of a test analyst at a computer. Four arrows point from the analyst to four circular tasks:</p> <ul style="list-style-type: none"> <li>Design the test cases for 5 requirements (red circle)</li> <li>Execute the designed test cases (purple circle)</li> <li>Log the defects and need to fail the related test cases (green circle)</li> <li>Re-test the defect, and re-execute the corresponding failed test case after the defect has been resolved (blue circle)</li> </ul> <p>In the above scenario, if metrics are not followed, then the work completed by the test analyst will be subjective. For example, the test report will not have the proper information to know the status of his work or project.</p>	

Slide Content	Use this space for your own notes
<p>Slide 52</p> <div data-bbox="405 421 925 812"><p>Significance of Software Testing Metrics (2 of 3) </p><p>If Metrics are involved in the project, then the exact status of work with proper numbers or data can be published.</p><p>In the test report, we can publish:</p><ul style="list-style-type: none"><li>1. How many test cases have been designed per requirement?</li><li>2. How many test cases are yet to be designed?</li><li>3. How many test cases are executed?</li><li>4. How many test cases are passed / failed / blocked?</li><li>5. How many test cases are not yet executed?</li><li>6. How many defects are identified and what is the severity of those defects?</li><li>7. How many test cases have failed due to one particular defect?</li></ul><p><small>Basic Testing © Copyright IBM Corporation 2015 </small></p></div> <p>Based on the project needs, we can have more metrics than the above mentioned list, to know the status of the project in detail.</p>	

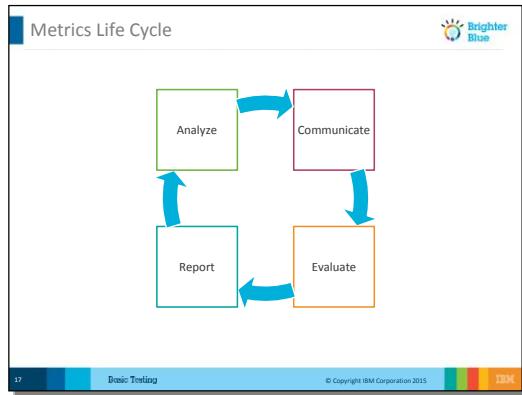
Slide Content	Use this space for your own notes
<p>Slide 53</p> <div data-bbox="405 421 925 812"><p>Significance of Software Testing Metrics (3 of 3) </p><p>Based on the metrics (on the previous slide), test lead or manager will get the understanding of the below mentioned key points:</p><ul style="list-style-type: none"><li>▶ Percentage (%) of work completed</li><li>▶ Percentage (%) of work yet to be completed</li><li>▶ Time to complete the remaining work</li><li>▶ Whether the project is going as per the schedule or lagging, and so on</li></ul><p><small>© Copyright IBM Corporation 2015 </small></p></div> <p>Based on the metrics, if the project is not going to complete as per schedule, then the manager will raise the alarm to the client and other stake holders by providing the reasons for lagging to avoid the last minute surprises.</p>	

Slide Content	Use this space for your own notes																																										
<p>Slide 54</p> <div data-bbox="403 421 925 812"><p>Example of Measurements</p><table border="1"><thead><tr><th>S. Number</th><th>Testing</th><th>Data retrieved during test case development and execution</th></tr></thead><tbody><tr><td>1</td><td>Number of Requirements</td><td>5</td></tr><tr><td>2</td><td>Avg. Number of Test cases written per Requirement</td><td>20</td></tr><tr><td>3</td><td>Total Number of Test cases written for all Requirements</td><td>100</td></tr><tr><td>4</td><td>Total Number of Test cases Executed</td><td>65</td></tr><tr><td>5</td><td>Number of Test cases Passed</td><td>30</td></tr><tr><td>6</td><td>Number of Test cases Failed</td><td>26</td></tr><tr><td>7</td><td>Number of Test cases Blocked</td><td>9</td></tr><tr><td>8</td><td>Number of Test cases not executed</td><td>35</td></tr><tr><td>9</td><td>Total Number of Defects identified</td><td>30</td></tr><tr><td>10</td><td>Critical Defects count</td><td>6</td></tr><tr><td>11</td><td>High Defects Count</td><td>10</td></tr><tr><td>12</td><td>Medium Defects Count</td><td>6</td></tr><tr><td>13</td><td>Low Defects Count</td><td>8</td></tr></tbody></table><p>Basic Testing © Copyright IBM Corporation 2015 IBM</p></div>	S. Number	Testing	Data retrieved during test case development and execution	1	Number of Requirements	5	2	Avg. Number of Test cases written per Requirement	20	3	Total Number of Test cases written for all Requirements	100	4	Total Number of Test cases Executed	65	5	Number of Test cases Passed	30	6	Number of Test cases Failed	26	7	Number of Test cases Blocked	9	8	Number of Test cases not executed	35	9	Total Number of Defects identified	30	10	Critical Defects count	6	11	High Defects Count	10	12	Medium Defects Count	6	13	Low Defects Count	8	
S. Number	Testing	Data retrieved during test case development and execution																																									
1	Number of Requirements	5																																									
2	Avg. Number of Test cases written per Requirement	20																																									
3	Total Number of Test cases written for all Requirements	100																																									
4	Total Number of Test cases Executed	65																																									
5	Number of Test cases Passed	30																																									
6	Number of Test cases Failed	26																																									
7	Number of Test cases Blocked	9																																									
8	Number of Test cases not executed	35																																									
9	Total Number of Defects identified	30																																									
10	Critical Defects count	6																																									
11	High Defects Count	10																																									
12	Medium Defects Count	6																																									
13	Low Defects Count	8																																									

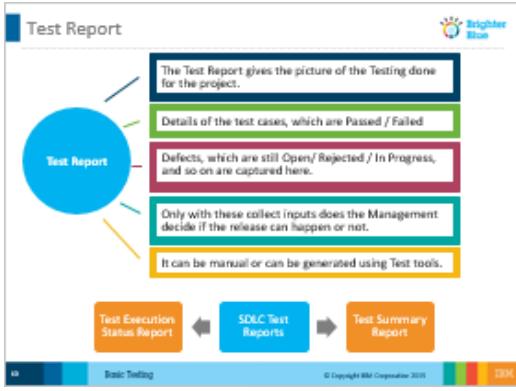
Slide Content	Use this space for your own notes				
<p>Slide 55</p> <div style="border: 1px solid #ccc; padding: 10px;"> <p><b>Metrics Examples</b></p> <p>Percentage (%) Test Cases Executed: This metric is used to obtain the execution status of the test cases in terms of Percentage (%).</p> <p>Percentage (%) Test cases Executed = (Number of Test cases executed / Total Number of Test cases written) * 100.</p> <p>So, from the above data,      Percentage (%) Test cases Executed  <math>= (65 / 100) * 100 = 65\%</math></p> <div style="text-align: center; margin-top: 10px;"> <p><b>Test execution completion percentage (%)</b></p>  <table border="1" style="margin-left: auto; margin-right: auto;"> <tr> <td>Number of Test Cases not executed</td> <td>35%</td> </tr> <tr> <td>Total Number of Test Cases Executed</td> <td>65%</td> </tr> </table> </div> <p style="font-size: small; color: #ccc;">IBM Basic Testing © Copyright IBM Corporation 2015</p> </div>	Number of Test Cases not executed	35%	Total Number of Test Cases Executed	65%	
Number of Test Cases not executed	35%				
Total Number of Test Cases Executed	65%				

Slide Content	Use this space for your own notes								
<p>Slide 56</p> <div data-bbox="405 421 925 812"> <p>Metrics Examples (continued)</p> <p>Percentage (%) Test cases Passed / Failed / Blocked</p> <p>Formula: Percentage (%) Test cases Passed / Failed / Blocked = (Number of Test cases Passed / Failed / Blocked / Total Number of Test cases Executed) * 100.</p> <p>So, from the above data,</p> <p>Percentage (%) Test cases Passed = <math>(30 / 65) * 100 = 46\%</math></p> <p>Percentage (%) Test cases Failed = <math>(26 / 65) * 100 = 40\%</math></p> <p>Percentage (%) Test cases Blocked = <math>(9 / 65) * 100 = 14\%</math></p> <p><b>Test execution status</b></p> <table border="1"> <thead> <tr> <th>Status</th> <th>Percentage</th> </tr> </thead> <tbody> <tr> <td>Number of Test Cases Blocked</td> <td>14%</td> </tr> <tr> <td>Number of Test Cases Failed</td> <td>40%</td> </tr> <tr> <td>Number of Test Cases Passed</td> <td>46%</td> </tr> </tbody> </table> <p>IS Basic Testing © Copyright IBM Corporation 2015 IBM</p> </div>	Status	Percentage	Number of Test Cases Blocked	14%	Number of Test Cases Failed	40%	Number of Test Cases Passed	46%	
Status	Percentage								
Number of Test Cases Blocked	14%								
Number of Test Cases Failed	40%								
Number of Test Cases Passed	46%								

Slide Content	Use this space for your own notes
<p>Slide 57</p> <div data-bbox="403 421 925 812"><p>Benefits of Metrics</p><p>Metrics helps in:</p><ul style="list-style-type: none"><li>Controlling the project</li><li>Monitoring the health of the project</li><li>Ensuring project objectives and the IBM strategy are met</li><li>Making data-driven decisions</li><li>Identifying areas to drive process improvements</li><li>Identifying point to do a root cause analysis</li><li>Making prediction</li></ul><p>IS Basic Testing © Copyright IBM Corporation 2015 IBM</p></div>	

Slide Content	Use this space for your own notes
<p>Slide 58</p>  <pre> graph TD     A[Analyze] --&gt; B[Communicate]     B --&gt; C[Evaluate]     C --&gt; D[Report]     D --&gt; A     </pre> <p>Analyze:</p> <ul style="list-style-type: none"> <li>▪ Identify the Test Metrics</li> <li>▪ Define the identified Metrics</li> </ul> <p>Communicate:</p> <ul style="list-style-type: none"> <li>▪ Explain the need of metrics to stakeholder and testing team.</li> <li>▪ Educate the testing team about the data points that need to be captured for processing the metric.</li> </ul> <p>Evaluate:</p> <ul style="list-style-type: none"> <li>▪ Capture and verify data.</li> <li>▪ Calculating the metric(s) value using the data captured.</li> </ul>	

Slide Content	Use this space for your own notes
<p>Generate:</p> <ul style="list-style-type: none"><li>▪ Develop the report with effective conclusion.</li><li>▪ Distribute report to the stakeholder and respective representative.</li><li>▪ Take feedback from stakeholder.</li></ul>	

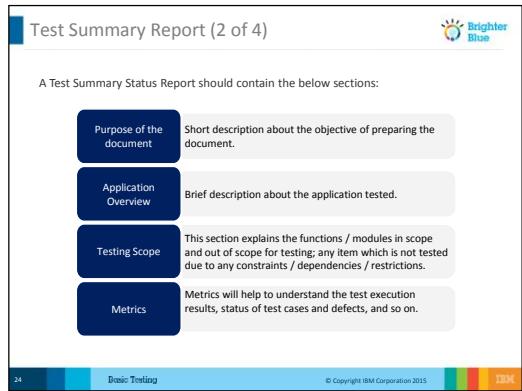
Slide Content	Use this space for your own notes
<p>Slide 59</p>  <p>The diagram illustrates the components of a Test Report and its role in the Software Development Life Cycle (SDLC). It features a central blue circle labeled "Test Report" connected by lines to four rectangular boxes:</p> <ul style="list-style-type: none"> <li>The top-left box is green and contains the text: "The Test Report gives the picture of the Testing done for the project."</li> <li>The top-right box is purple and contains: "Details of the test cases, which are Passed / Failed."</li> <li>The bottom-left box is red and contains: "Defects, which are still Open/Rejected / In Progress, and so on are captured here."</li> <li>The bottom-right box is yellow and contains: "Only with these collect inputs does the Management decide if the release can happens or not."</li> </ul> <p>Below these boxes, a horizontal flowchart shows the process: "Test Execution Status Report" (orange box) → "SDLC Test Reports" (blue box) → "Test Summary Report" (orange box). The "SDLC Test Reports" box has a double-headed arrow between it and each of the four boxes above it.</p> <p>At the bottom of the diagram, there are three colored bars: blue, teal, and orange, followed by the text "Basic Testing". On the right side, there is a small copyright notice: "© Copyright IBM Corporation 2010".</p> <p><b>Test reports used in the SDLC:</b></p> <ul style="list-style-type: none"> <li>▪ Test Execution Status Report</li> <li>▪ Test Summary Report</li> </ul>	

Slide Content	Use this space for your own notes
<p>Slide 60</p>  <p>The slide content shows a screenshot of a 'Test Execution Status Report (1 of 4)' from Brighter Blue. The report has a green header bar with the title and the Brighter Blue logo. The main body contains a green box labeled 'Test Execution Status Report' and a grey box with the following bullet points:</p> <ul style="list-style-type: none"><li>▪ This is a communication sent out to establish transparency to the QA team's activities of the day during the test cycle.</li><li>▪ This includes both defect information and test case run information.</li><li>▪ It is sent to Development, Environment support, Business analyst, and the Project teams.</li></ul> <p>The footer of the slide includes a navigation bar with icons for back, forward, and search, the text 'Basic Testing', and the IBM logo.</p>	

Slide Content	Use this space for your own notes
<p>Slide 61</p> <div data-bbox="403 421 925 812"><p>Test Execution Status Report (2 of 4) </p><p>Test Execution Status Report should contain the following <b>10 points</b>:</p><ul style="list-style-type: none"><li><input type="checkbox"/> Number of test cases planned for that day</li><li><input type="checkbox"/> Number of test cases executed that day</li><li><input type="checkbox"/> Number of test cases executed overall</li><li><input type="checkbox"/> Number of defects encountered that day and their respective states</li><li><input type="checkbox"/> Number of defect encountered so far and their respective states</li></ul><p>20 Basic Testing © Copyright IBM Corporation 2015 </p></div>	

Slide Content	Use this space for your own notes
<p>Slide 62</p> <div data-bbox="405 421 925 812"><p>Test Execution Status Report (3 of 4) </p><p>Number of critical defects that are still open</p><p>Environment downtimes, if any</p><p>Showstoppers, if any</p><p>Attachment of the test execution sheet / Link to the <a href="#">test management tool</a> where the test cases are placed</p><p>Attachment to the bug report / link to the defect / test management tool used for incident management</p><p>You can view an example of a Test Execution Report on the next slide.</p><p>23 Basic Testing © Copyright IBM Corporation 2015 </p></div>	

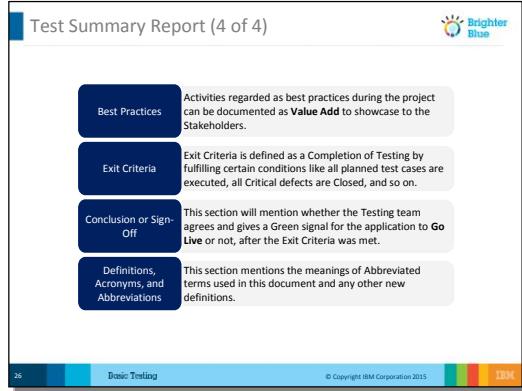
Slide Content	Use this space for your own notes
<p>Slide 63</p>  <p>The slide displays a screenshot of a 'Test Summary Report (1 of 4)' document. The document has a green header bar with the title 'Test Summary Report' and a blue footer bar with the text 'Basic Testing'. The main content area contains three bullet points explaining the purpose and scope of a Test Summary Report:</p> <ul style="list-style-type: none"><li>▪ Test Summary Report is an important deliverable which is prepared at the end of a Testing project, or rather after Testing is completed.</li><li>▪ The prime objective of this document is to explain various details and activities about the Testing performed for the project to the respective stakeholders like senior management, client, and so on.</li><li>▪ As part of Test execution report, daily testing results are shared with involved stakeholders every day. But Test Summary Report provides a consolidated report on the Testing performed so far for the project.</li></ul>	

Slide Content	Use this space for your own notes
<p>Slide 64</p>  <p>The slide displays a Test Summary Report (2 of 4) from Brighter Blue. It lists four sections: Purpose of the document (short description about the objective of preparing the document), Application Overview (brief description about the application tested), Testing Scope (explains functions/modules in scope and out of scope for testing), and Metrics (helps understand test execution results, status of test cases and defects). The slide footer includes page number 24, a navigation bar for 'Basic Testing', and the IBM logo.</p> <ul style="list-style-type: none"> <li>▪ <b>Purpose of the document:</b> Short description about the objective of preparing the document</li> <li>▪ <b>Application Overview:</b> Brief description about the application tested.</li> <li>▪ <b>Testing Scope:</b> This section explains about the functions / modules in scope and out of scope for testing; any items which are not tested due to any constraints / dependencies/ restrictions.</li> <li>▪ <b>Metrics:</b> Metrics will help to understand the test execution results, status of test cases and defects, and so on. Required Metrics can be added as necessary. For example: Defect Summary-Severity wise; Defect</li> </ul>	

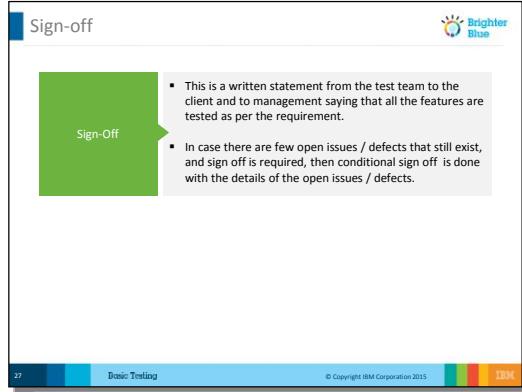
Slide Content	Use this space for your own notes
Distribution-Function/Module wise; Defect Ageing and so on, Charts / Graphs can be attached for better visual representation.	

Slide Content	Use this space for your own notes
<p>Slide 65</p>  <p>The slide displays a Test Summary Report (3 of 4) with the following sections:</p> <ul style="list-style-type: none"> <li><b>Types of testing performed:</b> Describe the various types of Testing performed in a project.</li> <li><b>Test Environment and tools:</b> Provide details on Test Environment in which the Testing is carried out; Server, Database, Application URL, and so on.</li> <li><b>Lessons Learned:</b> This section is used to describe the critical issues faced and their solutions (how they were solved during Testing).</li> <li><b>Recommendations:</b> Any workaround or suggestions can be mentioned here.</li> </ul> <p>At the bottom, there is a navigation bar with icons for Back, Forward, Home, and Help, along with the text "Basic Testing" and "© Copyright IBM Corporation 2015".</p> <ul style="list-style-type: none"> <li>▪ <b>Types of testing performed:</b> Describe the various types of Testing performed for the Project. This will make sure the application is being tested properly through testing types agreed as per Test Strategy.</li> <li>▪ <b>Test Environment and tools:</b> Provide details on Test Environment in which the Testing is carried out; Server, Database, Application URL and so on. If any Tools were used like Quality Center (now HP ALM) for logging defects.</li> <li>▪ <b>Lessons Learned:</b> This section is used to describe the critical issues faced and their solutions (how they were solved during the Testing). Lessons learned will help to make proactive decisions during the next Testing</li> </ul>	

Slide Content	Use this space for your own notes
<p>engagement, by avoiding these mistakes or finding a suitable workaround.</p> <ul style="list-style-type: none"><li>▪ <b>Recommendations:</b> Any workaround or suggestions can be mentioned here.</li></ul>	

Slide Content	Use this space for your own notes
<p>Slide 66</p>  <ul style="list-style-type: none"> <li>▪ <b>Best Practices:</b> There will be lot of activities done by the Testing team during the project. Some of them could have saved time, some proved to be a good and efficient way to work, and so on. These can be documented as a <b>Value Add</b> to showcase to the Stakeholders.</li> <li>▪ <b>Exit Criteria:</b> Exit Criteria is defined as a Completion of Testing by fulfilling certain conditions like:             <ol style="list-style-type: none"> <li>All planned test cases are executed;</li> <li>All Critical defects are Closed and so on.</li> </ol> </li> <li>▪ <b>Conclusion / Sign Off:</b> This section will mention whether the Testing team agrees and gives a Green signal for the application to <b>Go Live</b> or not, after</li> </ul>	

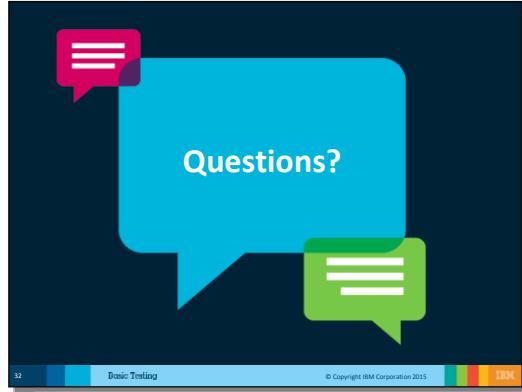
Slide Content	Use this space for your own notes
<p>the Exit Criteria was met. If the application does not meet the Exit Criteria, then it can be mentioned as the application is not suggested to <b>Go Live</b>. It will be left with the decision of Senior Management and Client and other Stakeholders involved to take the call on whether the application can <b>Go Live</b> or not.</p> <ul style="list-style-type: none"><li>▪ <b>Definitions, Acronyms, and Abbreviations:</b> This section mentions the meanings of Abbreviated terms used in this document and any other new definitions.</li></ul>	

Slide Content	Use this space for your own notes
<p>Slide 67</p>  <p>As we have to notify all the stakeholders that testing has begun, it is also the QA team's duty to let everyone know that testing has been complete and share the results. So, typically an email is sent from the QA team (usually the team lead / QA manager) giving an indication that QA team has signed off on the product attaching the test results and the list of open or known issues.</p>	

# Core Testing > Basic Testing > Day 9

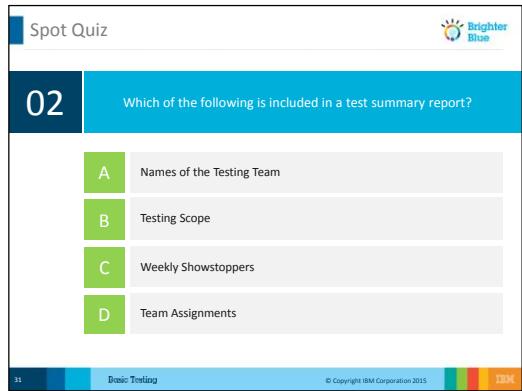
Slide Content	Use this space for your own notes
<p>Slide 68</p> <div data-bbox="405 421 925 812"><p>Sample Sign-off Email</p><p>To: Client, PM, Dev team, DB team, BA, QA team, Environment Team (and anyone else that needs to be included)</p><p>Hello Team,</p><p>The QA team signs-off on the Orange HRM version 3.0 software after the successful completion of the 2 cycles of functional testing the website.</p><p>The test cases and their execution results are attached to the email. (Or mention the location where they are present. If using test management software, provide details regarding the same.)</p><p>The list of known issues is attached to the email too. (Again, any other references that make sense can be added.)</p><p>Thanks, QA team lead.</p><p>28 Basic Testing © Copyright IBM Corporation 2015 IBM</p></div>	

Slide Content	Use this space for your own notes
<p>Slide 69</p>  <p>The slide features a yellow background with a blue sidebar on the left containing icons for a keyboard, a monitor, gears, and a hand holding a gear. At the top, the word "Activity" is written in a circular arrangement of colored circles. Below it, the text "Create a Sign-Off email" is displayed. A photograph of a person's hands typing on a laptop keyboard is shown. The bottom of the slide includes a navigation bar with a back arrow, the number "29", the title "Basic Testing", a copyright notice "© Copyright IBM Corporation 2015", and the IBM logo.</p> <ul style="list-style-type: none"><li>▪ Create your own sample sign-off email based on the example found in the previous slide.</li><li>▪ Make sure you include all the relevant details as what was discussed.</li></ul>	

Slide Content	Use this space for your own notes
<p>Slide 70</p>  <p>The slide features a dark blue background. In the center is a large light blue speech bubble containing the text "Questions?". To its left is a small pink speech bubble, and to its right is a small green speech bubble, both with horizontal lines inside. At the bottom of the slide is a navigation bar with several colored squares and the text "Basic Testing" and "Copyright IBM Corporation 2015".</p>	

Slide Content	Use this space for your own notes

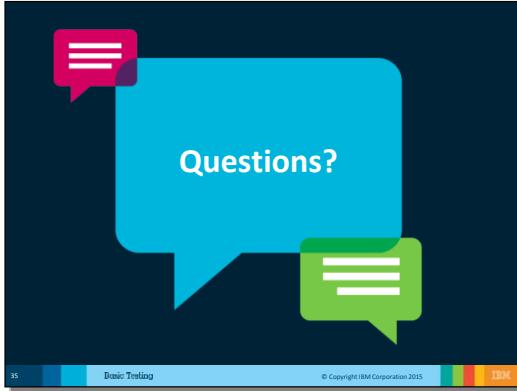
Slide Content	Use this space for your own notes
<p>Slide 71</p> <div data-bbox="405 421 925 812"><p>Spot Quiz</p><p>01 Which of the following is NOT a benefit of metrics?</p><p>A It helps in making data-driven decisions. B It helps in monitoring the health of a project. C It ensures how project objectives and the IBM strategy are met. D It prohibits the testing team to control the project.</p><p>10 Basic Testing © Copyright IBM Corporation 2015 IBM</p></div>	

Slide Content	Use this space for your own notes
<p>Slide 72</p>  <p>The slide displays a quiz interface with the title "Spot Quiz" and the question number "02". The question is "Which of the following is included in a test summary report?". The options are:</p> <ul style="list-style-type: none"><li>A Names of the Testing Team</li><li>B Testing Scope</li><li>C Weekly Showstoppers</li><li>D Team Assignments</li></ul> <p>At the bottom, there is a navigation bar with icons for back, forward, and search, followed by the text "Basic Testing" and "© Copyright IBM Corporation 2015".</p>	

## Module 04: Case Study

Slide Content	Use this space for your own notes
<p>Slide 73</p> 	

Slide Content	Use this space for your own notes
<p>Slide 74</p> <div data-bbox="386 421 903 812"><p>Case Study: Day 9—Defect Management and Reporting</p><p>Let us get started with some real life case studies now. Here is what you need to do:</p><ul style="list-style-type: none"><li>▪ Work with your team as per instructions from the facilitator</li><li>▪ Discuss the various reporting options of RQM and defect reports to analyze the status of the test project among the team</li><li>▪ The observer will note down the key points from the discussion</li><li>▪ Share your key takeaways with the class based on the discussion (30 mins)</li></ul><p> Microsoft Word 17 - 2003 Document</p><p>34 Basic Testing © Copyright IBM Corporation 2015 </p></div>	

Slide Content	Use this space for your own notes
Slide 75  <p>35 Basic Testing © Copyright IBM Corporation 2015 IBM</p>	