Exploratory Testing Explained

Exploratory Testing Explained”, www.satisfice.com/articles/et-article.pdf

Exploratory testing is also known as ad hoc testing.

What makes exploratory testing interesting, and in my view profoundly important,
is that when a tester has the skills to *listen*, *read, think* and *report,* rigorously and effectively, without the use of pre-scripted instructions, the exploratory approach to testing can be many times as productive (in terms of revealing vital information) as the scripted variety.

When I solve a jigsaw puzzle, I change how I work as I learn about the puzzle and see the picture form. If I notice a big blotch of color, I might decide to collect all the pieces of that approximate color into one pile. If I notice some pieces with a particularly distinctive shape, I might collect those together. If I work on one kind of testing for a while, I might switch to another kind just to keep my mind fresh. If I find I’ve got a big enough block of pieces assembled, I might move it into the frame of the puzzle to find where it connects with everything else. Sometimes I feel like I’m too disorganized, and when that happens, I can step back, analyze the situation, and adopt a more specific plan of attack. Notice how the process *flows*, and how it remains continuously, *each moment*, under the control of the practitioner. Isn’t this very much like the way you would assemble a jigsaw, too? If so, then perhaps you would agree that it would be absurd for us to carefully document these thought processes in advance. Reducing this activity to one of following explicit instructions would only slow down our work.

the puzzle changes the puzzling.

The power of exploratory tests can be
optimized throughout the test process, whereas scripts, because they don’t change, tend to become less powerful over time.

the chance that you
will find a problem on the second execution of the script is, in most circumstances, substantially lower than if you ran a new test instead.

Test Design: An exploratory tester is first and foremost a test designer. Anyone can design a test accidentally, the excellent exploratory tester is able to craft tests that systematically explore the product. That requires skills such as the ability to analyze a product, evaluate risk, use tools, and think critically, among others.
• *Careful Observation:* Excellent exploratory testers are more careful observers than novices, or for that matter, experienced scripted testers. The scripted tester need only observe what the script tells him to observe. The exploratory tester must watch for *anything* unusual or mysterious. Exploratory testers must also be careful to distinguish obervation from inference, even under pressure, lest they allow preconceived assumptions to blind them to important tests or product behavior.
• *Critical Thinking:* Excellent exploratory testers are able to review and explain their logic, looking for errors in their own thinking. This is especially important when reporting the status of a session of exploratory tests, or investigating a defect.
• *Diverse Ideas:* Excellent exploratory testers produce more and better ideas than novices. They may make use of heuristics to accomplish this. Heuristics are mental devices such as guidelines, generic checklists, mnemonics, or rules of thumb. The

Satisfice Heuristic Test Strategy Model (http://www.satisfice.com/tools/satisfice-tsm-4p.pdf) is an example of a set of heuristics for rapid generation of diverse ideas. James Whittaker and Alan Jorgensen's "17 attacks" is another. The diversity of tester temperaments and backgrounds on a team can also be harnessed by savvy exploratory testers through the process of group brainstorming to produce better test ideas.

- *Rich Resources:* Excellent exploratory testers build a deep inventory of tools, information sources, test data, and friends to draw upon. While testing, they remain alert for opportunities to apply those resources to the testing at hand.

Cem
Kaner suggests regular meetings with testers to discuss test progress, at least once per week. He finds it useful to open the meeting with a standard question, "What is the most interesting bug you've found recently? Show it to me."