

# What We Expect You to Know

- Random value generation: `runif()`, `rexp()`, `rbinom()`, `rgeom()`, ...
- Sample and shuffle integers using `sample.int()`
- Use `sample()` on vectors if you are *sure* they are not numerics
- Use `set.seed()` to initialize the PRNG and get reproducible results
- Use `RNGversion()` to make sure the same PRNG is used in different R versions
- Get the PRNG-state from `.Random.seed`; you can save it, and then restore it by assigning a saved value to `.Random.seed`.
  - With some random functions, you have to call `set.seed(dummyvalue)` before to reset them.
  - `.Random.seed` is *not* the same value as given to `set.seed()`.
- Changing your code (or even the version of a package) can have an influence on PRNG call order (and therefore your results) in surprising ways; consider it possible that all values change once something in the code in between has changed.
- Using cached values or conditionally skipping parts of your code will lead to differing PRNG states; cache the `.Random.seed`, or call `set.seed()` again after optional chunks, to avoid the problem
- Do not run different parts of your code with the same seed.