Monte Carlo particle transport applications are often written in low-level languages (C/C++) for optimal performance on clusters and supercomputers. However, this development approach often sacrifices straightforward usability and testing in the interest of fast application performance. To improve usability, some high-performance computing applications employ mixed-language programming with high-level and low-level languages. In this study, we consider the benefits of incorporating an interactive Python interface into a Monte Carlo application. With PyMercury, a new Python extension to the Mercury general-purpose Monte Carlo particle transport code, we improve application usability without diminishing performance. In two case studies, we illustrate how PyMercury improves usability and simplifies testing and validation in a Monte Carlo application. In short, PyMercury demonstrates the value of interactive Python for Monte Carlo particle transport applications. In the future, we expect interactive Python to play an increasingly significant role in Monte Carlo usage and testing.