We introduce a new query primitive called Function Query (FQ). An FQ operates on a set of math functions and retrieves the functions whose output with a given input satisfies a query condition (e.g., being among top k or within a given range). While FQ finds its natural uses in querying a database of math functions, it can also be applied on a database that originally stores discrete values. By interpreting the database as a set of user-defined functions, FQ can achieve the same functionality as existing analytic queries such as top-k query. Towards the goal of support FQ in a Database Management System, we address two challenges. First, we propose an efficient indexing technique for a database of functions, to support efficient execution of FQ. Second, we design an authentication algorithm, which allows a user to verify if an FQ is correctly executed by an untrustworthy Database-as-a-Service (DaaS) provider, such as Amazon Web Service. The two techniques together provide a framework to efficiently support FQ in system level for big data applications.