When should I use MongoDB instead of MySQL?

It all depends on what kind of data you are trying to store.

MySQL and MongoDB are two rather specific types of databases. Generally you first pick the storage method you want. The storage methods divide themselves into two groups: *relational* and *NoSQL*. And the *NoSQL* storage methodes can be divided in about four groups (there's a lot of division methods): *column store*, *key-value store*, *graph store*, *document store*. MongoDB is a document store database system.

But just to compare the two: a *relational database* is excellent for storing facts that all have intrinsic connections between them, that are the same every time. Let's say you have a school, and there's classes that have students, and students attend courses, and courses have teachers, who teach students, give out grades, et cetera. Every noun in that sentence could be made into a table, with fixed, intrinsic relations with the other tables. A student never teaches classes, a teacher never receives grades, and so on. You could still make a relational database that allows students to teach classes, using association tables, but if that keeps happening you'll probably want to have a different kind of database altogether.

NoSQL databases in general are for when the data you are storing does not have (that many) fixed and intrinsic inner relations. Say, you want to store your error logs. A log has a code, a message, a level, and so on. But those are all attributes of your log, not separate entities with many-to-many relations with other entities.

Now, one of the other store methods I mentioned is *key-value store*. That means getting rid of all connections between data. You've got just a single row with a key and your value(s) (using JSON you can store as many values in a single field as you want). *Document store* can be called a variant of that, except it accommodates more complex values more easily.

So, you can store data in MySQL if it is relational (connections have many intrinsic connections to other entities), and in MongoDB if it is not relational at all, and the values are rather complex. But really you should decide on your store system of choice before picking the individual database. MongoDB and MySQL are the most popular databases of their kind, but other systems may work better for you for more detailed reasons than I could mention here.

Link:

https://www.quora.com/When-should-we-use-MongoDB-and-MySQL-together