Подпрограмма данных - это программная единица, которая предназначена для ввода данных (вроде оператору DATA), которые размещаются в спальных зонах памяти, для дотупу из различных программных единиц. Как правило, это большие массивы данных, организовать доступ к которым по-другому будет слишком сложно.
Подпрограмма данных начинается оператором BLOCK DATA, содержит операторы описания типов данных, оператору описания общих блоков памяти, операторы DATA для задания значений. Подпрограмма данных завершается оператором END (END BLOCK DATA).
Как и подпрограмма, она имеет собственное имя, по которому может быть идентифицирована, но с которым не связано никакого значения (числа, символов и т.п.). Учитывая ее назначения, подпрограмма данных не может быть вызвана из других программных единиц, а также с ней не имеет смысла вызывать другую программную единицу.

Пример. Подпрограмма данных. Типичные начало и окончание.
BLOCK DATA DAT
...
END BLOCK DATA DAT

Модуль - специфическая программная единица, введенная в Фортран для поддержки возможностей объектно ориентированного программирования. Основное назначение модуля - расширение внешне видимости объектов.
По парадигмой процедурного программирования, программные единицы имеют собственные локальные объекты, зона видимости которых ограничена программной единицей. Они не видны за ее пределами. Данные таких объектов можно передать в другой программной единицы, но ее имя залишится локальным. Для явно объявленных в модуле объектов можно расширить зону видимости имен на выбранные или на все программные единицы. Таким образом, если есть возможность обращаться к объектам по их именам, то отпадает необходимость в программных усилиях по организации передачи данных этих объектов между программными единицами.
В силу таких особенностей, к моменту обычной компиляции программных единиц, модули должны пройти препроцессорную обработку. Поэтому в проекте они должны быть размещены или в отдельном файле (файлах), или в файле с основной программой, причем, перед ее началом.
Модуль начинается оператором MODULE, может иметь любые операторы, заканчивается оператором END (END MODULE). Он имеет собственное имя, по которому может быть идентифицирован, но с которым не связано никакого значения (числа, символов, др.). Модуль не может быть вызван из других программных единиц, однако может быть асоцийований в основной программе, подпрограмме, функции или другом модуле через оператор USE.

Пример. Составить модуль MOD для расширенной зоны видимости переменных ALPHA, BETA, массива DELTA (2, 20).
MODULE MOD

```
REAL ALPHA, BETA
DIMENSION DELTA (2, 20)
END MODULE
```

В программной единицы, где планируется использование переменных ALPHA, BETA, массива DELTA (вместе всех или некоторых из них), необходимо ассоциации к модулю MOD через оператор USE.

```
USE MOD
```