

功能测试命令文档

本文档是对功能测试中需要解析的命令的定义。

注意：阅读此文当前请仔细阅读并理解需求说明文档和测试说明文档，确保理解本文档中的所有内容，如有疑问询问助教，否则出现由于对命令的误解而造成测试结果偏差的情况都将由各小组自己负责。

功能测试命令的基本形式

模块+选项+操作: (model + option + action)

- 1. 模块分为球员和球队：分别用-player 和-team 表示。
- 2. 选项是可选的，可能有一个或者多个 option，但是也可能整条命令都不含有 options。如果有多个，则 option 之间用单个空格隔开，出现顺序按照下文中排列的顺序即可。
- 3. 操作主要包括两种类型，分别为过滤、排序。

各模块中功能测试命令的选项和操作

- 1. 球员模块(player)

选项(option):

-avg 和-total 是一组选项，不可以同时出现，如果都没有出现，则默认为-avg。

-avg: 如果有这一项，则返回结果中包含的数据应为平均数，举例来说，如球员得分这项应该返回球员的场均得分。

-total: 如果有这一项，则表示返回的结果中包含的数据应为总和，相应的就球员得分来说应返回球员在所有比赛中的得分总和。

示例如下：

-player -avg

-player

上面两个命令效果一样

-all、-hot field 和-king field -season/daily 是一组选项，不可以同时出现，如果都没有出现，则默认为-all。

-all: 获得所有球员的数据。

-hot field: 获得热门球员的数据，如果有这一项，就不会有-avg、-total 选项和-filter、-sort 操作，数据都为-avg 形式，field 表示获得热门球员的依据，其可能值如下所示：

Model	field
Player	score: 得分
	rebound: 篮板
	assist: 助攻

-king field -season/-daily: 获得赛季/当日数据王的信息，-avg、-total 选项和-filter、-sort 操作不会与此选项同时出现，数据都为-avg 形式，field 表示获得数据王的依据，其可

能值如下所示：

Model	field
Player	score: 得分
	rebound: 篮板
	assist: 助攻

示例如下：

-player -king score -season: 获得赛季得分王

-player -king assist -daily: 获得当日数据王

-n number:这个选项有两部分，需要同时写入，之间用单个空格隔开，**number** 处应填入一个正整数，表示获得符合查询条件的数据中的前 **number** 条，例如-n 10 则表示返回前 10 个球员的相关信息；如果 **number** 的数目大于所有的结果的数量，则返回所有的结果。如果没有这一项，则 **number** 默认为 50。

-high: 获得球员的高阶数据，如果没有这一项，则所有返回的数据是基本数据，有这一项，则返回的都是高阶数据，能与这一项同时出现的只有-n 选项和-sort 操作。

操作(action):

-filter field.value: 这个操作用来过滤掉部分数据源，其中 **field** 表示过滤的依据，**value** 表示 **field** 预期的值，而且 **field.value** 对可能出现多次，之间用单个“,”隔离，出现顺序按照下面表格中的排列顺序即可，**field** 的可能值及相应 **value** 可能的值（大小写固定，冒号及以后为解释说明）如下：

field	Value
position: 球员位置	F: 前锋
	G: 后卫
	C: 中锋
	All: 所有的（默认）
league: 球员联盟	West: 西部联盟
	East: 东部联盟
	All: 所有的（默认）
age: 球员年龄	<=22: 小于等于 22 岁
	22< X <=25: 大于 22 岁小于等于 25 岁（注意空格）
	25< X <=30: 大于 25 岁小于等于 30 岁
	>30: 大于三十岁
	All: 所有的（默认）

如果没有-filter 这一项，则使用默认的过滤器过滤。

示例如下：

-player -filter position.F,league.West : 只在西部前锋中挑选

-sort field.sortOrder: 这一选项决定按哪种属性及什么排序规则进行排序，其中 **field** 是决定排序顺序的属性，**sortOrder** 决定排序的顺序，**sortOrder** 的值分为 **asc** 和 **desc**，分别表示升序排列和降序排列，**sortOrder** 不要省略，而且 **field.sortOrder** 对可能出现多次，之间用单个“,”隔离，表示按照多个属性排序，位置越靠前的 **field.sortOrder** 优先级越高，**field** 可能的值（大小写固定）如下：

model	dataType	field
player	基础数据	point: 得分
		rebound: 篮板
		assist: 助攻
		blockShot: 盖帽
		steal: 抢断
		foul: 犯规
		fault: 失误
		minute: 分钟（上场时间）
		efficient: 效率
		shot: 投篮命中率
		three: 三分命中率
		penalty: 罚球命中率
		doubleTwo: 两双
	高阶数据	realShot: 真实投篮命中率
		GmSc: GmSc 效率值
		shotEfficient: 投篮效率
		reboundEfficient: 篮板效率
		offendReboundEfficient: 进攻篮板率
		defendReboundEfficient: 防守篮板率
		assistEfficient: 助攻率
		stealEfficient: 抢断率
		blockShotEfficient: 盖帽率
		faultEfficient: 失误率
		frequency: 使用率

如果没有这一项，则基础数据默认为-sort score.desc，高阶数据默认为-sort realShot.desc

注：对于排序属性的值相同的球员按球员姓名的升序排列。

2. 球队模块(team):

选项(option):

-avg 和-total 是一组选项，不可以同时出现，如果都没有出现，则默认为-avg。

-avg: 如果有这一项，则返回结果中包含的数据应为平均数，举例来说，如球队得分这项应该返回球队的场均得分。

-total: 如果有这一项，则表示返回的结果中包含的数据应为总和，相应的就球队得分来说应返回球队在所有比赛中的得分总和。

-all 和-hot field 是一组选项，不可以同时出现，如果都没有出现，则默认为-all。

-all: 获得所有球队的数据。

-hot field: 获得热门球队的数据，如果有这一项，就不会有-avg、-total 选项和-sort 操作，数据都为-avg 形式，field 表示获得热门球队的依据，其可能值如下所示：

model	dataType	field
team	基础数据	score: 得分
		rebound: 篮板

		assist: 助攻
		blockShot: 盖帽
		steal: 抢断
		foul: 犯规
		fault: 失误
		shot: 投篮命中率
		three: 三分命中率
		penalty: 罚球命中率
		defendRebound: 防守篮板数
		offendRebound: 进攻篮板数
		rebound: 篮板数

-n number:这个选项有两部分，需要同时写入，之间用单个空格隔开，number 处应填入一个正整数，表示获得符合查询条件的数据中的前 number 条，例如-n 10 则表示返回前 10 个球队的相关信息；如果 number 的数目大于所有的结果的数量，则返回所有的结果。如果没有这一项，则 number 默认为 30。

-high: 获得球队的高阶数据，如果没有这一项，则所有返回的数据是基本数据，有这一项，则返回的都是高阶数据，能与这一项同时出现的只有-n 选项和-sort 操作。

操作(action):

-sort field.sortOrder: 这一选项决定按哪种属性及什么排序规则进行排序，不可省略，其中 field 是决定排序顺序的属性，sortOrder 决定排序的顺序，sortOrder 的值分为 asc 和 desc，分别表示升序排列和降序排列，field 可能的值（大小写固定）如下：

model	dataType	field
team	基础数据	point: 得分
		rebound: 篮板
		assist: 助攻
		blockShot: 盖帽
		steal: 抢断
		foul: 犯规
		fault: 失误
		shot: 投篮命中率
		three: 三分命中率
		penalty: 罚球命中率
		defendRebound: 防守篮板数
		offendRebound: 进攻篮板数
		rebound: 篮板数
	高阶数据	winRate: 胜率
		offendRound: 进攻回合
		offendEfficient: 进攻效率
		defendEfficient: 防守效率
		offendReboundEfficient: 进攻篮板效率

		defendReboundEfficient: 防守篮板效率
		stealEfficient: 抢断效率
		assistEfficient: 助攻效率

如果没有这一项，则基础数据默认为-sort score.desc，高阶数据默认为-sort winRate.desc。

注：对于排序属性的值相同的球队按球队名的升序排列。

功能测试命令的具体示例

model	Command	explanation
player	-player	返回得分前 50 的球员的场均比赛数据
	-player -all -n 10	返回得分前 10 的球员的总和比赛数据
	-player -high -n 10 -sort frequency.desc	返回使用率前 10 的的球员的高阶数据
	-player -hot assist -n 5	返回助攻数前 5 的热门球员
	-player -king score -season	返回前 5 的赛季得分王
	-player -avg -n 5 -filter position.F	返回前锋中得分前 5 的球员的场均比赛数据
	-player -total -all -n 10 -filter position.F,league.west - sort shot.desc	返回西部联盟中投篮命中率最高的 10 名前锋的总和比赛数据
team	-team	返回得分前 30 的球队的场均比赛数据
	-team -all -n 10	返回得分前 10 的球队的总和数据
	-team -hot assist -n 5	返回助攻数前 5 的热门球队
	-team -avg -n 5 -sort shot.asc	返回投篮命中率最差的 5 支球队的场均比赛数据
	-team -total -all -n 10 -sort shot.desc	返回命中率前 10 的球队的总和数据
	-team -high -n 5 -sort stealEfficient.asc	返回抢断效率最差的 5 支球队的高阶数据