



У цій діаграмі є три сутності: **Game**, **GameTeam** і **Team**. **Game** містить поле **teams**, яке є масивом об'єктів типу **GameTeam**, що містять інформацію про команди, які беруть участь у грі. Кожен об'єкт **GameTeam** має ключ **game** і **team**, які посилаються на об'єкти **Game** і **Team** відповідно. Крім того, об'єкти **Team** містять масив **games**, що містить посилання на об'єкти типу **GameTeam**, які містять цю команду.


Таким чином, ця діаграма показує, що є взаємозв'язок між сутностями **Game**, **GameTeam** і **Team**, який дозволяє зберігати інформацію про гри, команди і зв'язки між ними в базі даних.

Games			...
 _id	ObjectId	NN	
date	String	NN	
teams[ ]	ObjectId	NN	
team1Score	Number	NN	
team2Score	Number	NN	



GameTeam			...
 _id	ObjectId	NN	
game	ObjectId	NN	
team	ObjectId	NN	



Team			...
 _id	ObjectId	NN	
name	String	NN	
games[ ]	ObjectId	NN	

```
var mongoose = require('mongoose');
var Schema = mongoose.Schema;
var Games = new Schema({
  date: {
    type: String,
    required: true,
    match: /\d{4}-\d{2}-\d{2}/
  },
  teams: [{
    type: Schema.Types.ObjectId,
    required: true,
    ref: 'GameTeam'
  }],
  team1Score: {
    type: Number,
    required: true,
    min: 0,
    max: 10
  },
  team2Score: {
    type: Number,
    required: true,
    min: 0,
    max: 10
  }
});
```

```
var mongoose = require('mongoose');
var Schema = mongoose.Schema;
var GameTeam = new Schema({
  game: {
    type: Schema.Types.ObjectId,
    required: true,
    ref: 'Game'
  },
  team: {
    type: Schema.Types.ObjectId,
    required: true,
    ref: 'Team'
  }
});
```

```
var mongoose = require('mongoose');
var Schema = mongoose.Schema;
var Team = new Schema({
  name: {
    type: String,
    required: true,
    minlength: 3,
    maxlength: 20
  },
  games: [{
    type: Schema.Types.ObjectId,
    required: true,
    ref: 'GameTeam'
  }]
});
```