



- [Home](#)
- [Simulations](#)
- [Odds Oracle](#)
- [PQL](#)
- [Videos](#)
- [Forum](#)
- [Welcome to PQL](#) [PQL Runner](#) [What is PQL? \(Help\)](#) [Sample PQL Queries](#)

PQL Documentation

- [Overview](#)
- [Selectors](#)
- [From clause](#)
- [Where clause](#)
- [Functions](#)
- [Special Syntax \(handsHaving\)](#)
- [Types](#)

Poker Query Language Overview

What is PQL?

PQL is a language meant for people who want to ask poker probability questions. It is the core engine that powers the Odds Oracle and propokertools.com.

What kinds of questions can PQL answer?

All kinds of questions.

Can you give an example?

Sure. If we want to know the average all-in equity of aces vs a random hand in holdem, we would write the following PQL query:

```
select avg(riverEquity(hero)) as heroEquity from game='holdem', hero='AA', villain='**'
```

OK, but the example above could have been achieved with the existing ProPokerTools Simulator! Got anything better?

Let's suppose we want to know how often a hand with two aces in omaha which starts as a favorite on the flop gets outdrawn on the turn by a random hand, but then goes on to win the river. We can answer that with the following query:

```
select count(HvHequity(villain, turn) > 0.5 and winsHi(hero)) as suckAndResuck
from game='omahahi', hero='AA**', villain='*****'
where HvHequity(hero, flop) > 0.5
```

Not bad. What else ya got?

The following query will tell you how often A2 gets quartered (or sixthed) in omaha-8 against two decent hands when it makes the nut low by the river:

```
select count(tiesLo(hero)) as lostSomeChips
from game='omaha8', hero='A2**', villain1='15%', villain2='15%'
```

where nutLo(hero, river) and not (winsHi(hero) or tiesHi(hero))

Hey, this is kind of fun!

I'm glad you think so! Here's another one. This will show you what hand types (pair, two pair, etc.) win in stud in a 5-handed random race:

```
select histogram(winningHandType())
```

```
from game='studhi', syntax='generic', p1='*', p2='*', p3='*', p4='*', p5='*'
```

What games does PQL support?

'holdem', 'omahahi', 'omaha8', 'studhi', 'stud8', 'razz', 'omahahi5' (5-card omaha hi), 'omaha85' (5-card omaha-8)

Why did you call it PQL?

The syntax of PQL is loosely based on SQL (Structured Query Language) which is the industry standard for querying databases. PQL uses a subset of the SQL operators and syntax - AND, OR, NOT, ||, <> (not equal), =, >, >=, <, <=, SELECT, FROM, WHERE, AS, *, /, -, +, CASE/WHEN/THEN/ELSE/END, IN, line comments (starting with -- and ending with a newline), block comments (start with /* and end with */), and parentheses for grouping.

Who is the target audience for PQL?

PQL is intended for people with a decent level of technical and/or poker sophistication who nevertheless do not have the time or inclination to write custom programs or mathematical proofs. It is also intended for people who enjoy the process of exploring poker probability for its own sake.

Selectors

PQL supports the following selectors:

- avg - The 'avg' selector computes averages of numbers. For example 'avg(riverEquity(p1))'.
- count - The 'count' selector counts how often a given boolean expression evaluates to true. For example 'count(wins(p1))', 'count(handType(p1, river) = pair)', 'count(HvHequity(p1, flop) > 0.5 and HvHequity(p2, flop) > 0.3)'
- histogram - The 'histogram' selector takes an expression that returns one of equity, hand types, player counts, equity fractions, card counts, or strings, and shows how often each possible outcome occurs. For example 'histogram(riverEquity())', 'histogram(handtype(p2, river))', 'histogram(handshaving(minhandtype, flop, pair))', 'histogram(outsToHandType(p1, flop, straight))'
- max - The 'max' selector computes the maximum value of an expression. For example, 'max(handType(hero, flop))'
- min - The 'min' selector computes the minimum value of an expression. For example, 'min(fractionalRiverEquity(villain))'

A PQL query supports any number of selectors separated by commas, and selectors can be given names using an optional 'as' clause. For example:

```
select count(wins(hero)) as heroWon,
avg(riverEquity(hero)) as heroEV,
histogram(handtype(hero, fifth)) as fifthStreetHandType
from ...
```

From Clause

The PQL from clause allows for the specification of the game (required), the board (if any), the dead cards (if any), the syntax (if needed), and the range of hands for each player. Syntax defaults to 'generic' if not

specified. Here is an example showing all of the options in a from clause:

```
select ...
from game='holdem', syntax='generic', board='KsJhTd', dead='2c',
hero='AsAd', looseyGoosey='**', tightOne='AA-TT, AK, AQ'
```

Where Clause

The PQL where clause is an optional boolean expression that must hold true before any selectors are evaluated. It can be used to specify the situation more tightly than is possible with the from clause alone.

For example:

```
select ...
from ...
where HvHequity(villain, flop) > HvHequity(hero, flop)
AND handtype(hero, turn) >= pair
```

Functions

Below you will find a list of the functions currently supported in PQL. If you have a function you'd like us to add, feel free to [email us your suggestion](#)

| Function Name | Argument Types | Return Type | Description |
|----------------------------|------------------|-------------|---|
| bestHiRating | TPlayer, TStreet | TBoolean | Returns true if no other hand has a better hi on the given street. For instance, where p1='AK', p2='JT', board='AsKsQhJdTc', bestHiRating(p1, flop) is true, bestHiRating(p2, turn) is false, and bestHiRating(p2, river) is true. |
| bestLoRating | TPlayer, TStreet | TBoolean | Returns true if we have a lo and no other hand has a better lo on the given street. For instance, in stud-8 where p1='2356798', p2='23568qa', bestLoRating(p1, fifth) is true, bestHiRating(p1, sixth) is true, and bestHiRating(p1, seventh) is false. |
| boardAllowsMadeLo | TStreet | TBoolean | Returns true if a made low is possible on the given street. |
| boardHasOneDistinctLoCard | TStreet | TBoolean | Returns true if there is exactly one distinct lo card on the given street. |
| boardHasTwoDistinctLoCards | TStreet | TBoolean | Returns true if there are exactly two distinct lo cards on the given street. |

| | | | |
|----------------------|------------------|------------|---|
| boardInRange | TBoardRange | TBoolean | <p>Returns true if the board is in the given range. For instance, if the board is 'AsKdJhKc3s', then boardInRange('AKK') is false, boardInRange('AKJ3K') is false, and boardInRange('KJAK3') is true. Note that card order is only relevant on the turn and river.</p> |
| boardLoCardCount | TStreet | TCardCount | <p>Returns the number of distinct lo cards on the board for the given street without regard to suit. For instance, if the board is 'Ks8c7c7s2d', distinctLoBoardCardCount(flop) = 2, distinctLoBoardCardCount(turn) = 2, and distinctLoBoardCardCount(river) = 3</p> |
| boardRanks | TStreet | TRankSet | <p>Returns the ranks that are on the board for the given street</p> |
| boardSuitCount | TStreet | TCardCount | <p>Flop games only. Returns the number of different suits on the board for a given street. For instance, if board='KsJsTc9d8s', boardSuitCount(flop) = 2, boardSuitCount(turn) = 3, and boardSuitCount(river) = 4</p> |
| cardsPlay | TPlayer, TStreet | TCardCount | <p>Returns the number of cards (2, 1, or 0) in a player's hold'em hand that play on the turn or river. For instance, on KsJsTd4c2d with KdJd 2 cards play on the river, whereas with Kd3c only one card plays on the river (the king). If a card is duplicated on the board, it is not considered to play; on AsKsQdJdTc with JhTh 0 cards play on the river.</p> |
| duplicatedBoardRanks | TStreet | TRankSet | <p>Returns a set of all ranks that are duplicated on the given street.</p> |
| duplicatedHandRanks | TPlayer, TStreet | TRankSet | <p>Returns a set of all ranks that appear more than once for the given player on the given street. For flop games, the answer will</p> |

| | | | |
|-----------------------|-----------------------------|-------------------|--|
| equity | TPlayer, TStreet | TEquity | be the same for all streets. Short for HvHEquity Evaluates to true if a player has a particular flop hand category. See <code>flopHandCategory()</code> . Evaluates to true if a player has a certain type of hand on a given street. For instance, <code>exactHandType(playerOne, flop, pair)</code> evaluates to true if player one has a pair on the flop. Returns a number between 1 and 7,462 for a given player and street based on the absolute strength of the best 5-card hand. For instance, if player one holds AsKs and the board is QsJsTs, this returns 1, whereas Ks9s would return 2, 9s8s would return 3, As9s would return 4, and so on. |
| exactFlopHandCategory | TPlayer, TFlopHandCategory | TBoolean | Returns the flop hand category for a particular player. This gives more specific information than the <code>handType</code> function. For instance, if P1 has AK on a JJ4 flop, his <code>handType</code> on the flop is one pair (jacks), but his <code>flopHandCategory</code> is nothing, since he missed the flop |
| exactHandType | TPlayer, TStreet, THandType | TBoolean | Flop games only. Evaluates to true if a flush is possible. With a board of ks7d4s7s3s, <code>flushingBoard(flop)</code> is false, <code>flushingBoard(turn)</code> is true and <code>flushingBoard(river)</code> is true. |
| fiveCardHiHandNumber | TPlayer, TStreet | TInteger | Flop games only. Given a player and a street (one of flop or turn), evaluates to true if the player has at least one four flush. For example, <code>fourFlush(p1, flop)</code> would return true for a holdem hand of AhKh and a board of JhTh9s. Note that for hold'em, at least one of the player's cards must play. |
| flopHandCategory | TPlayer | TFlopHandCategory | |
| flushingBoard | TStreet | TBoolean | |
| fourFlush | TPlayer, TStreet | TBoolean | |

| | | | |
|------------------------|------------------|--------------|---|
| fractionalRiverEquity | TPlayer | TFraction | <p>Returns the fraction of the pot a player wins on the river, such as 2/3 or 1/4</p> <p>Flop games only. Computes the number of cards in a player's hand that have the same rank as a card on the board. If p1='AsKd' and the board is '2d8s9dAhKs', then boardIntersections(p1, flop) = 0, boardIntersections(p1, turn) = 1, and boardIntersections(p1, river) = 2.</p> |
| handBoardIntersections | TPlayer, TStreet | TCardCount | |
| handRanking | TPlayer | THandRanking | <p>Returns a handRanking for the default (full-ring) ranking. See handRankingFor().</p> <p>Returns a hand ranking between 1 and 100 for a given ranking name with 1 being the best. For instance, in hold'em, handRankingFor(p1, '10h') = 1 if p1 = 'AA', and handRanking(p1, '10h') = 100 if p1='7s2c'</p> |
| handRankingFor | TPlayer, TString | THandRanking | |
| handRanks | TPlayer, TStreet | TRankSet | <p>Returns the hand ranks for the given player on the given street. For flop games, the answer will be the same for all streets.</p> <p>Gives the hand type for a given player and street. For instance, given playerOne='KsKhJdJs2c6d9h', handType(playerOne, fifth) evaluates to pair</p> |
| handType | TPlayer, TStreet | THandType | |
| hasSecondBoardRank | TPlayer, TStreet | TBoolean | <p>Returns true if the given player has the second-highest board rank on the given street. Returns false if only one rank on the board.</p> <p>Returns true if the given player has at least one card matching the largest rank on the board for the given street.</p> |
| hasTopBoardRank | TPlayer, TStreet | TBoolean | |
| hiRating | TPlayer, TStreet | THiRating | <p>Hi and hi/lo games only. Returns a rating indicating the relative strength of the best hi hand. Useful for comparing hand strengths throughout a hand. This</p> |

| | | | |
|------------------------|--------------------------|---------|---|
| | | | <p>rating has no meaning on its own - as such it is only useful when compared to other hands in the same game. For instance, in stud, where p1='2s5s9stsjdQsAh', and p2='7h7d9c9d3s3c3d', <code>hiRating(p1, fifth) < hiRating(p2, fifth)</code>, <code>hiRating(p1, sixth) > hiRating(p2, sixth)</code>, and <code>hiRating(p1, seventh) < hiRating(p2, seventh)</code>.</p> |
| HvHEquity | TPlayer, TStreet | TEquity | <p>Flop games only. Computes exact hand vs. hand equity for a given player on a given street. For example, <code>HvHEquity(playerOne, flop)</code> gives the equity for playerOne's hand on the flop vs. every one else's hand on the flop.</p> |
| HvPerceivedRangeEquity | TPlayer, TStreet, TRange | TEquity | <p>Computes the randomized heads-up hand vs. range equity on the given street for the given player against a perceived range using max 10000 trials and max 10 seconds. For instance, <code>HvPerceivedRangeEquity(p1, flop, 'AA-TT')</code> gives the equity for the specific hand dealt to p1 on the flop against the range 'AA-TT'. For flop games, the random hands and complete board are generated, the board is trimmed to the appropriate size for the given street, and a hand vs. range simulation is performed.</p> |
| HvREquity | TPlayer, TStreet | TEquity | <p>Computes the randomized hand vs. range equity on the given street for the given player using max 10000 trials and max 10 seconds. For instance, <code>HvREquity(p1, flop)</code> gives the equity for the specific hand dealt to p1 on the flop against the ranges given for every other player. For flop games, the random hands and complete</p> |

| | | | |
|-----------------------|------------------|-----------|---|
| | | | <p>board are generated, the board is trimmed to the appropriate size for the given street, and a hand vs. range simulation is performed. For stud games, each player is dealt a hand in their range, then cards are added to bring their hand size up to the given street, with face-up cards for street 4-6 and face down for street 7. Then, a hand vs range simulation is performed with the specified player's hand vs the opponents hand ranges with the additional up-cards added.</p> |
| inRange | TPlayer, TRange | TBoolean | <p>Evaluates to true if the hand dealt to a certain player falls within a given range. For instance, given playerOne = 'KK', inRange(playerOne, 'K*') evaluates to true, while inRange(playerOne, 'QJ') evaluates to false. Range weights (aka 'AK@10,AQ@2') are ignored in this function.</p> |
| intersectingHandRanks | TPlayer, TStreet | TRankSet | <p>Returns a RankSet of all ranks in the given player's hand that match the board on the given street.</p> |
| loRating | TPlayer, TStreet | TLoRating | <p>Lo and hi/lo games only. Returns a rating indicating the relative strength of the best lo hand. If no lo is possible, a rating lower than all other low ratings is returned. Useful for comparing hand strengths throughout a hand. This rating has no meaning on its own - as such it is only useful when compared to other hands in the same game. For instance, in stud8, where p1='Ac3c5c7c8c9c2c', and p2='As3s5s7sTc2sJs', loRating(p1, fifth) > loRating(p2, fifth), loRating(p1, sixth) < loRating(p2, sixth), and loRating(p1, seventh) =</p> |

| | | | |
|---------------------------|-----------------------------------|----------|--|
| madeLo | TPlayer, TStreet | TBoolean | <p>loRating(p2, seventh).</p> <p>Given a player and a street (one of flop, turn, or river), evaluates to true if the player made a lo. For example, madeLo(p1, flop) would return true where p1='A399' and board='7s6c4d'</p> |
| maxRank | TRankSet | TRank | <p>Returns the maximum rank in a set. Will fail on an empty rank set.</p> |
| minEquity | TPlayer, TStreet, TDouble | TBoolean | <p>Synonym for minHVHEquity.</p> |
| minFlopHandCategory | TPlayer, TFlopHandCategory | TBoolean | <p>Evaluates to true if a player has a certain minimum type of flop hand category. See flopHandCategory().</p> |
| minHandType | TPlayer, TStreet, THandType | TBoolean | <p>Evaluates to true if a player has a certain minimum type of hand on a given street. For instance, minHandType(playerOne, flop, pair) evaluates to true if player one has at least a pair on the flop.</p> |
| minHiRating | TPlayer, TStreet, THiRating | TBoolean | <p>Returns true if the player equals or beats the given hand rating on the given street. For instance, minHiRating(player, flop, someHandRating)</p> |
| minHvHEquity | TPlayer, TStreet, TDouble | TBoolean | <p>Flop games only. Evaluates to true if a player has a certain minimum amount of hand vs. hand equity on a given street. For instance, minHVHEquity(playerOne, flop, 0.5) evaluates to true if player one's hand has at least 50% equity on the flop against the other hands dealt on the flop.</p> |
| minHvPerceivedRangeEquity | TPlayer, TStreet, TRange, TDouble | TBoolean | <p>Returns true if a certain player has a minimum amount of equity vs. a perceived range. See HvPerceivedRangeEquity</p> |
| minHvREquity | TPlayer, TStreet, TDouble | TBoolean | <p>Evaluates to true if a player has a certain minimum amount of hand vs. hand equity on a given street. For instance, minHvREquity(playerOne, flop,</p> |

| | | | |
|--------------------------|---------------------------------------|----------|---|
| | | | 0.5) evaluates to true if player one's hand has at least 50% equity on the flop against the other hand ranges (not hands) on the flop. |
| minLoRating | TPlayer, TStreet, TLoRating | TBoolean | Returns true if the player equals or beats the given hand rating on the given street. For instance, minLoRating(player, flop, someHandRating) |
| minOutsToHandType | TPlayer, TStreet, THandType, TInteger | TBoolean | Flop games only. Given a player, a street (one of flop or turn), a handType, and a number of outs, evaluates to true if the player has at least that many outs to make the given hand type. See outsToHandType. |
| minRank | TRankSet | TRank | Returns the minimum rank in a set. Will fail on an empty rank set. |
| monotoneBoard | TStreet | TBoolean | Flop games only. Returns true if the board has only one suit. Equivalent to 'suitCount(street) = 1' |
| nonIntersectingHandRanks | TPlayer, TStreet | TRankSet | Returns a RankSet of all ranks in the given player's hand that do not match the board on the given street. |
| nthRank | TInteger, TRankSet | TRank | Returns rank number N from a rank set, starting from the top. For instance, if the board is 'AsKdJh', then nthRank(1, boardRanks(flop)) = toRank('A'), nthRank(2, boardRanks(flop)) == toRank('K'), and nthRank(3, boardRanks(flop)) = toRank('J') |
| nutHi | TPlayer, TStreet | TBoolean | Flop games only. Evaluates to true if a given player has the nuts on a given street. For instance, nutHi(p1, flop) returns true for p1='Qs5s', board='JsKsAs'. It is assumed that any dead cards are known (so that nutHi(p1, flop) for p1='As2s', board='KsQsJs' is false, but adding dead='Ts' would make it true). |

| | | | |
|------------------|-----------------------------|------------|--|
| nutHiForHandType | TPlayer, TStreet | TBoolean | <p>Flop games only. Returns true if the given player has the best possible hand for his hand type. For instance, nutHiForHandType(p1, flop) returns true for p1 = AsKh and flop = AdTd2d because p1 has the best possible hand with handType onepair.</p> |
| nutHiOuts | TPlayer, TStreet | TCardCount | <p>Flop games only. Returns the number of nut hi outs on the given street. For instance, in hold'em with p1='KsJs' and board='QdTd4c', nutHiOuts(p1, flop) = 6.</p> |
| nutLo | TPlayer, TStreet | TBoolean | <p>Flop games only. Given a player and a street (one of flop, turn, or river), evaluates to true if the player made the lo. For example, nutLo(p1, flop) would return true where p1='A399' and board='7s6c2d', but false for board='7s6c2d3h'</p> |
| nutLoOuts | TPlayer, TStreet | TCardCount | <p>Flop games only. Returns the number of nut lo outs on the given street. For instance, in omaha hi/lo with p1='A299' and board='Qd8c4c8d', nutLoOuts(p1, turn) = 15.</p> |
| outsToHandType | TPlayer, TStreet, THandType | TCardCount | <p>Flop games only. Given a player, a street (one of flop or turn), and a handType, evaluates to the number of cards that would give the player that hand type on the next card that is better than the current hand. For example, outsToHandType(p1, flop, straight) with p1='JsTh' and board='9h8s2c' would evaluate to 8 (if 7s were a dead card, it would evaluate to 7 instead of 8.). outsToHandType(p1, turn, straight) with p1='Th9h' and board='8h7h2c3c' would evaluate to 6 (because two of the 'straight' outs gives straight-flush instead of straight). outsToHandType(p1,</p> |

| | | | |
|---------------|------------------|------------|---|
| | | | flop, straight) with p1='Td9d' and board ='8d7d2d' evaluates to 0, because p1 already beats a straight. |
| overpair | TPlayer, TStreet | TBoolean | Returns true if the given player's hand has at least two cards of the same rank that are larger than the largest rank on the board. |
| pairedBoard | TStreet | TBoolean | Flop games only. Evaluates to true if at least two cards have the same rank on the board. With a board of ks7d4c7h3s, pairedBoard(flop) is false, pairedBoard(turn) is true and pairedBoard(river) is true. |
| pocketPair | TPlayer | TBoolean | Returns true if the given player's hand has at least two cards of the same rank. |
| rainbowBoard | TStreet | TBoolean | Flop games only. Evaluates to true if no suit is repeated on the board. With a board of ks7d4c7s3s, rainbowBoard(flop) is true, but rainbowBoard(turn) is false. |
| rankCount | TRankSet | TCardCount | Returns the number of ranks in a RankSet |
| rateHiHand | TString | THiRating | Given a string of five cards, returns a Hi rating that can be used to compare with other Hi ratings |
| rateLoHand | TString | TLoRating | Given a string of five cards, returns a Lo rating that can be used to compare with other Lo ratings rateLoHand uses the rules of the game in determining the lo. . |
| riverCard | | TCard | Returns the card on the river |
| riverEquity | TPlayer | TEquity | Computes hand vs. hand equity after the last card has been dealt. |
| scoops | TPlayer | TBoolean | scoops(playerOne) evaluates to true if playerOne wins the entire pot. |
| straightBoard | TStreet | TBoolean | Flop games only. Evaluates to true if a straight is possible. With a board of Js9s2c8d7s, |

| | | | |
|-----------------|------------------|-----------|---|
| threeFlush | TPlayer, TStreet | TBoolean | <p>straightBoard(flop) is false, straightBoard(turn) is true, and straightBoard(river) is true.</p> <p>Flop games only. Given a player and a street, evaluates to true if the player has at least one three-flush. For example, threeFlush(p1, flop) would return true for a hold'em hand of AhKh and a board of Jh2c3d. Note that for hold'em, at least one of the player's cards must play.</p> |
| tiesHi | TPlayer | TBoolean | <p>tiesHi(playerOne) evaluates to true if playerOne ties for the high half of the pot with at least one other player.</p> |
| tiesLo | TPlayer | TBoolean | <p>tiesLo(playerOne) evaluates to true if playerOne ties for the low half of the pot with at least one other player.</p> |
| toCard | TString | TCard | Parses a single card string into a Card. For instance, toCard('As') |
| toRank | TString | TRank | Given a rank string, such as 'A', returns a Rank. |
| toString | Type | TString | Returns the string representation of the argument passed. |
| turnCard | | TCard | Returns the card on the turn |
| twoToneBoard | TStreet | TBoolean | <p>Flop games only. Returns true if the board has exactly two suits. Equivalent to 'suitCount(street) = 2'</p> |
| upCard | TPlayer, TStreet | TCard | Returns the up-card for the given player and street. |
| winningHandType | | THandType | <p>Returns the hand type of the winning hand. For instance, winningHandType() with a board of 'KsQsJs2c4d' and handone='KK' and handtwo='5s6s' would evaluate to flush.</p> |
| winsHi | TPlayer | TBoolean | <p>winsHi(playerOne) evaluates to true if playerOne wins the entire high half of the pot.</p> |

| | | | |
|--------|---------|----------|---|
| winsLo | TPlayer | TBoolean | winsLo(playerOne) evaluates to true if playerOne wins the entire low half of the pot. |
|--------|---------|----------|---|

handsHaving

PQL has a special higher-order function called 'handsHaving'. Given a boolean function that takes a player as its first argument, and values for all of the other arguments, it returns the number of players for which the function evaluated to true. A couple of examples should give you the idea:

- handsHaving(minHandtype, river, flush) would evaluate to the number of players with a flush or better on the river.
- handsHaving(minHvREquity, flop, 0.3) would evaluate to the number of players with at least 30% Hand vs. Range equity on the flop.
- handsHaving(tiesHi) would evaluate to the number of players who tied for the hi half of the pot.

Types

PQL types are relatively simple and you probably don't need to know much about them, but for completeness, here they are.

- TBoardRange - a range of hands for the board, eg 'AxKxJy'
- TBoolean - true or false
- TCard - a single card, such as the Jack of Diamonds
- TCardCount - an integer between 0 and 52
- TDouble - a double precision floating point number, such as 0.123
- TEquity - a TDouble between 0.0 and 1.0
- TFlopHandCategory - what was made on the flop. This provides more specific information than THandType. Values in order:
 - flopnothing: A hand that made nothing, but possibly has some draws
 - flopunderpair: A pocket pair lower than the lowest rank on the flop
 - flopthirdpair: A hand that hits the third (lowest) rank on the flop
 - floppocket23: A pocket pair between the second and third rank on the flop
 - flopsecondpair: A hand that hits the second rank on the flop. For instance, AJ on KJ4 or KKJ
 - floppocket12: A pocket pair between the first and second rank on the flop
 - floptoppair: A hand that hits the top rank on the flop. For instance, AK on KJ4 or KJJ
 - flopoverpair: A pocket pair larger than the largest rank on the flop
 - flopbottomtwo: A hand that hits the bottom two ranks of a flop with three different ranks
 - floptopandbottom: A hand that hits the top and bottom ranks of a flop with three different ranks
 - floptoptwo: A hand that hits the top two ranks on a flop with three different ranks
 - floptrips: A hand that makes three of a kind on a paired board
 - flopset: A hand that makes three of a kind on an unpaired board
 - flopstraight: A hand that makes a straight
 - flopflush: A hand that makes a flush
 - flopfullhouse: A hand that makes a full house. This takes precedence over pocket-pair types.
 - flopquads: A hand that makes four of a kind
 - flopstraightflush: A hand that makes a straight flush

- TFraction - 1/2, 2/5, 13/914...
- THandType - a 5-card poker hand. Values in order:
 - highcard
 - pair
 - twopair
 - trips
 - straight
 - flush
 - fullhouse
 - quads
 - straightflush
- THiRating - a hi-hand rating - only useful when comparing to other THiRating
- TInteger, TLong - a whole number, such as 2, 3, 5, 7 or 11.
- TLoRating - a lo-hand rating - only useful when comparing to other TLoRating. In games where there is a qualifier, "no low" will have a TLoRating lower than every other TLoRating.
- TNumeric - base type for all numbers
- TPlayer - one of the players specified in the from clause, such as player1 or hero
- TPlayerCount - an integer between 0 and the number of players specified
- TRange - a range of hands, such as 'AA-TT'
- TRank - a Rank (an ace, a ten, a deuce, etc.)
- TRankSet - a Set of unique Ranks
- TStreet - for flop games, one of preflop, flop, turn, or river. For stud games, one of third, fourth, fifth, sixth, or seventh
- TString - a string in quotes, such as 'hello'

[Contact us](#) · [Privacy Policy](#) · [Blog](#)

Follow @propokertools

Copyright © 2005-2013 ProPokerTools. All rights reserved.